ADC_Single_Channel_1 for KIT_AURIX_TC297_TFT

ADC single channel conversion

AURIX™ TC2xx Microcontroller Training
V1.0.1

Please read the Important Notice and Warnings at the end of this document

Scope of work

The Versatile Analog-to-Digital Converter (VADC) is configured to measure an analog signal using background scan request.

An analog input channel is continuously converted using the background scan mode. The input value is determined using the microcontroller's supply voltage, ground level or letting the analog pin open and floating. Three LEDs are used, each indicating a voltage interval. Thus depending on the conversion value, a certain LED will light up.

Copyright © Infineon Technologies AG 2020. All rights reserved.

Introduction

- The AURIX™ microcontrollers provide a series of analog input channels (up to 8 for each ADC) connected to a cluster of Analog/Digital Converters (up to 11) using the Successive Approximation Register (SAR) principle. Each converter of the ADC cluster is represented as a group and can operate independently of the others.

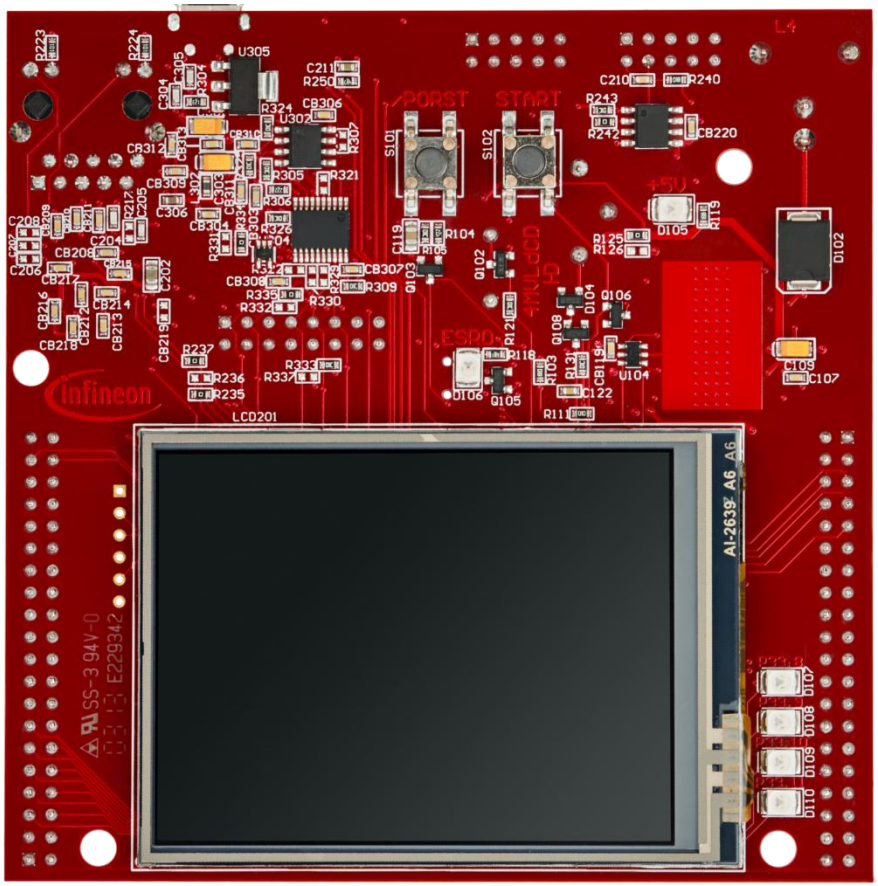
- Analog/Digital conversions can be requested by several request sources such as the Queued, Scan or Background Scan Source.

- Several conversion modes (Fixed Channel Conversion, Auto Scan Conversion or Channel Sequence Conversion), either executed as single or continuous, can be configured and used concurrently by the available request source. An arbiter resolves concurrent conversion requests from different sources.

Copyright © Infineon Technologies AG 2020. All rights reserved.

Hardware setup

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



Copyright © Infineon Technologies AG 2020. All rights reserved.

Implementation

- An application data structure defined in ***ADC_Single_Channel.h*** is used in this example, representing configuration values and references to setup and control the VADC.

  - A global object called ***g_vadcBackgroundScan*** from type ***ApplicationVadcBackgroundScan*** needs to be instantiated with the following elements:
    - ***vadc*** (from type ***IfxVadc_Adc***): references to the VADC module registers
    - ***adcGroup*** (from type ***IfxVadc_Adc_Group***): references to the VADC group registers
    - ***adcChannelConfig*** (from type ***IfxVadc_Adc_ChannelConfig***): stores the configuration values for the input channel
    - ***adcChannel*** (from type ***IfxVadc_Adc_Channel***): is a subset of ***adcChannelConfig*** and stores the channel id, the result register number and a reference to the VADC group

- The following functions, defined in ***ADC_Single_Channel.c***, are used:
  - ***vadcBackgroundScanInit()***
  - ***vadcBackgroundScanRun()***
  - ***indicateConversionValue()***
  - ***initializeLEDs()***

Copyright © Infineon Technologies AG 2020. All rights reserved.

Implementation

Configuring the VADC to run in background scan mode, continuously converting the analog input channel values and indicate them by LEDs

- The initialization is done through the functions ***vadcBackgroundScanInit()*** and ***vadcBackgroundScanRun()***.
  - The function ***vadcBackgroundScanInit()***:
    - creates a VADC configuration object ***adcConfig***
      - it is configured with default ADC configuration values using the function ***IfxVadc_Adc_initModuleConfig()***
      - the object is used to initialize the ADC (e.g. enable ADC, set ADC Power Supply, etc.) using the function ***IfxVadc_Adc_initModule()***
    - creates a group object ***adcGroup***
      - the group object is initialized with default values for each request source using the function ***IfxVadc_Adc_initGroupConfig()***
        - **e.g. the ADC to be used is specified, the ADC resolution is set to 12 bit, the sample time is set to 1 us**
      - signal to the arbiter that the group is using background scan
      - the object is used to initialize the group registers

Copyright © Infineon Technologies AG 2020. All rights reserved.

Implementation

- Furthermore, the function ***vadcBackgroundScanRun()*** does the following:
  - initializes the channel configuration object ***adcChannelConfig*** of the global defined application data structure ***g_vadcBackgroundScan*** with default values using the function ***IfxVadc_Adc_initChannelConfig()***
  - specifies the input channel number to be used for conversion by setting ***channelId***
  - specifies the result register to be used by setting the corresponding number to ***resultRegister***
  - specifies the channel as ***backgroundChannel***
  - initializes the channel object of the global defined application data structure ***g_vadcBackgroundScan***
  - starts background scan using the function ***IfxVadc_Adc_startBackgroundScan()***

Copyright © Infineon Technologies AG 2020. All rights reserved.

Implementation

- The visualization with LEDs is done using the functions ***initializeLEDs()*** and ***indicateConversionValue()***.
  - The function ***initializeLEDs()***
    - initializes the port pins 13.0, 13.1 and 13.2 as push-pull outputs using the function ***IfxPort_setPinMode()***
    - set the port pins 13.0, 13.1 and 13.2 to high state in order to switch the LEDs off by calling the function ***IfxPort_setPinHigh()***
  - The function ***indicateConversionValue()*** is continuously executed and
    - defines an object ***conversionResult*** from type ***Ifx_VADC_RES***
    - uses the function ***IfxVadc_Adc_getResult()*** to continuously retrieve the result value until the **valid flag** of the object ***conversionResult*** turns to high signaling that a new measurement is available
    - lights up the LED D107 (P13.0) if the discrete converted value is greater than 0xC00
    - lights up the LED D108 (P13.1) if the discrete converted value is smaller and equal than 0xC00 and greater and equal than 0x300
    - lights up the LED D109 (P13.2) if the discrete converted value is smaller than 0x300

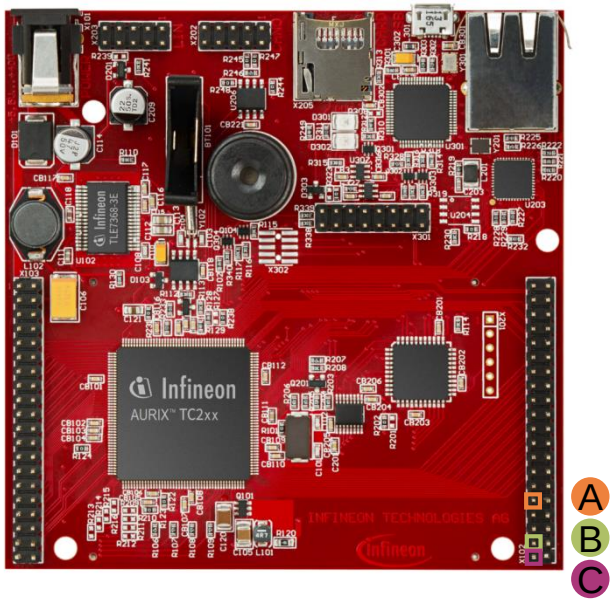
Copyright © Infineon Technologies AG 2020. All rights reserved.

Run and Test

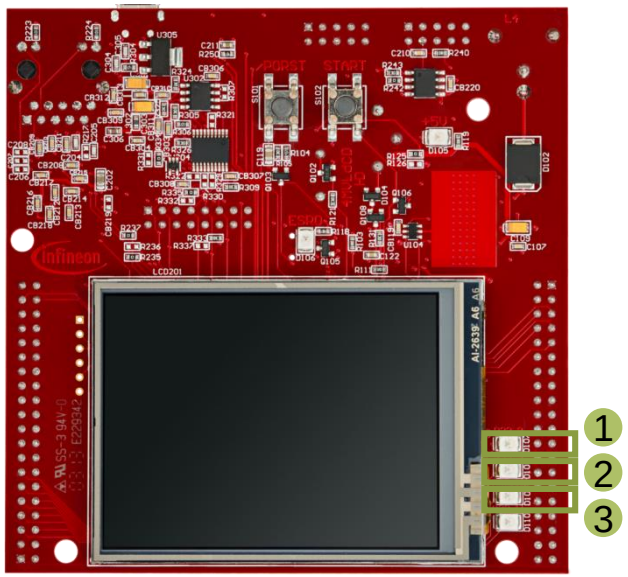
After code compilation and flashing the device, verify the behavior of the LEDs:

› Connect the pins according to the table and observe the LEDs (1), (2) and (3).

Connection	LED
leave (A) floating	(2) on, (1) (3) off *
(A) connected with (B)	(3) on, (1) (2) off
(A) connected with (C)	(1) on, (2) (3) off



	X102		
P14.5	40	39	P14.4
P20.10	38	37	P20.9
P15.7	36	35	P15.6
P15.5	34	33	P15.4
P15.3	32	31	P15.2
P22.3	30	29	P22.2
P22.1	28	27	P22.0
P33.11	26	25	P23.4
P23.3	24	23	P23.2
P23.1	22	21	P23.0
P33.6	20	19	P33.8
P33.12	18	17	P33.1
P33.2	16	15	P33.3
P33.4	14	13	P33.5
AN0	12	11	AN8
AN2	10	9	AN3
AN32	8	7	AN33
AN20	6	5	AN21
GND	4	3	GND
V_UC(+5V)	2	1	VCC_IN



*__Note:__ Depending on the surrounding noise, the floating port pin case may result in lighting up another LED (most general case is still (2)).

Copyright © Infineon Technologies AG 2020. All rights reserved.

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

Copyright © Infineon Technologies AG 2020. All rights reserved.

Revision history

Revision	Description of change
V1.0.1	Update of version to be in line with the code example’s version
V1.0.0	Initial version

Copyright © Infineon Technologies AG 2020. All rights reserved.

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

Edition 2020-12
Published by
Infineon Technologies AG
81726 Munich, Germany

© 2020 Infineon Technologies AG.
All Rights Reserved.

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

Document reference
ADC_Single_Channel_1_
KIT_TC297_TFT

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.