SCU_Emergency_Stop_1 for KIT_AURIX_TC375_LK

Emergency Stop via SCU

AURIX™ TC3xx Microcontroller Training
V1.0.1

Scope of work

This example shows how to trigger an emergency stop via an external signal and how port pins can be set to a defined state in this case.

The LED1, which is driven by the port pin P00.5, is blinking until an external signal triggers an emergency stop and sets the pin to emergency stop mode.

Introduction

- The System Control Unit (SCU) contains miscellaneous control registers associated with other functions such as controlling Application Test Mode and chip identification
- The Emergency Stop (EMS) is one of these functions. It provides a fast reaction to an emergency without the intervention of the software
- An emergency stop can be triggered by a transition on the port pin state which is configured as the EMS input or by an alarm event
- The Emergency Stop control logic for the port pins can operate in two modes:
  - Synchronous Mode: emergency case is activated by hardware and released by software (default and used in this training)
  - Asynchronous Mode: emergency case is activated and released by hardware

Hardware setup

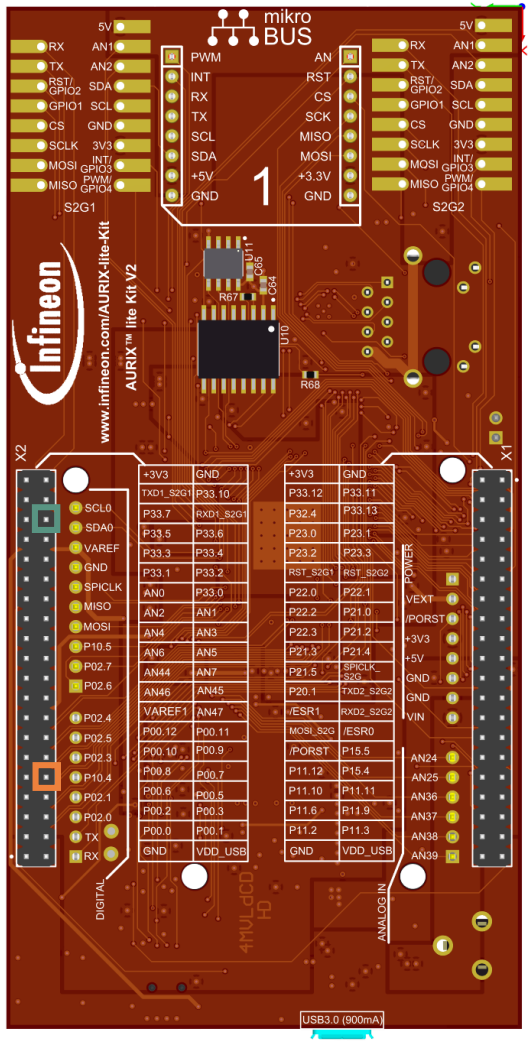
This code example has been developed for the board KIT_A2G_TC375_LITE.

Connect the emergency stop port pin P33.8 to the port pin P00.7 via a jumper.

		X2		
	+3V3	39	40	GND
TXD1_S2G1 -	P33.9	37	38	P33.10
	P33.7	35	36	P33.8 - RXD1_S2G1
	P33.5	33	34	P33.6
	P33.3	31	32	P33.4
	P33.1	29	30	P33.2
Potentiometer -	AN0	27	28	P33.0
	AN2	25	26	AN1
	AN4	23	24	AN3
	AN6	21	22	AN5
	AN44	19	20	AN7
	AN46	17	18	AN45
	VAREF1	15	16	AN47
	P00.12	13	14	P00.11
	P00.10	11	12	P00.9
	P00.8	9	10	P00.7 - Button1
LED2 -	P00.6	7	8	P00.5 - LED1
	P00.2	5	6	P00.3
	P00.0	3	4	P00.1
	GND	1	2	VDD_USB

Emergency stop

Button



Implementation

Configuring System Control Unit

Configuration of the System Control Unit (SCU) is done once in the setup phase by calling the initialization function ***initScuEmergency()***, which contains the following steps:

- Call the iLLD function ***IfxScuWdt_clearSafetyEndinitInline()*** to disable the Safety Endinit protection in order to modify the SCU register
- Set ***SCU_EMSR.B.POL*** to 0x0 to set input state as active high
- Set ***SCU_EMSR.B.MODE*** to 0x0 to select the synchronous mode
- Set ***SCU_EMSR.B.PSEL*** to 0x0 to select port A (pin P33.8) as emergency stop input
- Set ***SCU_EMSR.B.ENON*** to 0x1 to enable emergency stop flag
- Call the iLLD function ***IfxScuWdt_setSafetyEndinitInline()*** to re-enable the Safety Endinit protection

The functions ***IfxScuWdt_clearSafetyEndinitInline()*** and ***IfxScuWdt_setSafetyEndinitInline()*** are contained in the iLLD header ***IfxScuWdt.h***, while ***initScuEmergency()*** function is contained in ***SCU_Emergency_Stop.h***.

Implementation

Configuring port pin

Configuration of the port pins for emergency stop input and for the LED are also done in the function ***initScuEmergency()*** with the following steps:

- Call the iLLD function ***IfxPort_setPinMode()*** with ***IfxPort_Mode_inputPullDown*** as parameter to configure the emergency stop pin as input
- Call the iLLD function ***IfxPort_setPinMode()*** with ***IfxPort_Mode_outputPushPullGeneral*** as parameter for the input to configure the LED as output
- Enable the emergency stop for the LED with the function ***IfxPort_setESR()***

Toggling the LED

The LED is toggled in the function ***toggleLED()***, which contains a call of the iLLD function ***IfxPort_togglePin()***.

All of the above functions, called inside ***initSCUEmergency()*** and ***toggleLED()***, are contained in the iLLD header ***IfxPort.h***.

Run and Test

After code compilation and flashing the device, perform the following steps:

- Observe the LED1 (1) which should be blinking
- Switch the emergency pin state P33.8 by pressing the button (2)
- Observe the LED1 (1), which should be off

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

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

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

Edition 2021-06
Published by
Infineon Technologies AG
81726 Munich, Germany

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

Document reference
SCU_Emergency_Stop_1
_KIT_TC375_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.