

Teaching online electronics, microcontrollers and programming in Higher Education

# Programing of embedded systems

7. A/D Converter

Lead Partner: Warsaw University of Technology

**Authors: Daniel Krol** 

University of Applied Sciences in Tarnow



Declaration

This report has been prepared in the context of the ENGINE project. Where other published and unpublished source materials have been used, these have been acknowledged.

# Copyright

#### © Copyright 2021 - 2023 the ENGINE Consortium

Warsaw University of Technology (Poland)

International Hellenic University (IHU) (Greece)

European Lab for Educational Technology- EDUMOTIVA (Greece)

University of Padova (Italy)

University of Applied Sciences in Tarnow (Poland)

All rights reserved.



This document is licensed to the public under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.

# **Funding Disclaimer**

This project has been funded with support from the European Commission. This report reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

# **Programing of embedded systems**

7. A/D Converter

#### I. OLED Display

- 1. Create a new project for the LPCXpresso804 board and name it eg Lab07.
- 2. Add ADC, CTIMER and I2C drivers:

_					
ect name: ZPSW_Lab07		Project name suffix: Se	earch		
Use default location					
cation: /Users/daniel/Documents/	MCUXpressoIDE_11.5.0_7232/workspace_LPC55s69/ZPS	SW_Lab07			Browse.
evice Packages	Board	Project Type	Project Type		
LPC804M101JDH24 LPC804M101JDH20 LPC804M111JDH24	<ul> <li>Default board files</li> <li>Empty board files</li> </ul>	C Project C+ C Static Library C+	C Project C ++ Project C Static Library C++ Static Library		st 🧿 UART
omponents				omponents selection summary	Ē
			- · ·		
dd or remove SDK software compo	nents			type to filter	
dd or remove SDK software compo	nents IS Drivers Utilities Middleware Board Components A	bstraction Layer Software Component	ts]	type to filter	
dd or remove SDK software compo	nents IS Drivers ] Utilities ] Middleware ] Board Components ] A	bstraction Layer Software Component		type to filter Description	Versio Info
dd or remove SDK software compo perating Systems Drivers CMS Drivers	nents IS Drivers   Utilities   Middleware   Board Components   A	bstraction Layer Software Component		type to filter lame Description Description	Versio Info
dd or remove SDK software compo perating Systems Drivers CMS Drivers (type to filter	nents iIS Drivers \ Utilities \ Middleware \ Board Components \ A	bstraction Layer ेे Software Component 😥 ह		type to filter lame Description Descriptio	Versio Info
dd or remove SDK software compo perating Systems Drivers CMS Drivers (type to filter Name	nents IS Drivers Utilities Middleware Board Components A	bstraction Layer Software Component 🖉 ଶ୍		type to filter lame Description > ≣ Drivers > © Operating Systems > Utilities	Versio Info
do r remove SDK software compo perating Systems Drivers CMS Drivers (type to filter Name	nents IS Drivers   Utilities   Middleware   Board Components   A Description ADC Driver	bstraction Layer Software Component 교 왕 Version 2.5.0	ts Ts The Construction of the Construction o	type to filter lame Description > ≣ Drivers > ≣ Operating Systems > ≣ Utilities	Versio Info
dd or remove SDK software compo perating Systems Drivers CMS Drivers CMS Type to filter Name Seadc	nents IS Drivers   Utilities   Middleware   Board Components   A Description ADC Driver CAPT Driver	bstraction Layer Software Component		type to filter lame Description Description Description Description Description Description Description Description	Versio Info
d or remove SDK software compo perating Systems Drivers CMS Drivers type to filter Name Sadc Sadc Sadc	nents IS Drivers   Utilities   Middleware   Board Components   A Description ADC Driver CAPT Driver Clock Driver	bstraction Layer Software Component Version 2.5.0 2.10 2.3.1	ts N	type to filter iame Description > ≣ Drivers > ≣ Optivers > ≣ Utilities	Versio Info
d or remove SDK software compo perating Systems Drivers CMS Drivers type to filter Name Capt Capt Capt Capt Capt Capt Capt Capt	nents IS Drivers   Utilities   Middleware   Board Components   A Description ADC Driver CAPT Driver Clock Driver COMMON Driver	bstraction Layer Software Component	ts	type to filter lame Description ≥ Drivers > Goerating Systems > Utilities	Versio Info
d or remove SDK software compo perating Systems Drivers CMS Drivers type to filter Name Compared adc Compared adc Compared adc Compared adc Compared add Compared	IS Drivers Utilities Middleware Board Components A Description ADC Driver CAPT Driver Clock Driver COMMON Driver CTimer Driver	bstraction Layer Software Component Version 2.5.0 2.1.0 2.3.1 2.3.1 2.2.1		type to filter iame Description ≥ Dorivers ≥ Goperating Systems ⇒ Utilities	Versio Info
d or remove SDK software compo perating Systems Drivers CMS Drivers Cype to filter Name Compared Context Compared Context Compared Context Compared Context Compared Context C	IS Drivers   Utilities   Middleware   Board Components   A Description ADC Driver CAPT Driver Clock Driver COMMON Driver CTimer Driver DAC Driver	bstraction Layer Software Component ✓ § Version 2.5.0 2.1.0 2.3.1 2.3.1 2.2.1 2.0.2		type to filter Name Description > ≣ Drivers > ≣ Operating Systems > ≣ Utilities	Versio Info
d or remove SDK software compo perating Systems Drivers CMS Drivers (type to filter Name C C C C C C C C C C C C C C C C C C C	nents IS Drivers   Utilities   Middleware   Board Components   A Description ADC Driver CAPT Driver COMMON Driver COMMON Driver DRC Driver GPID Driver GPID Driver GPID Driver	bstraction Layer Software Component		type to filter lame Description ▷ Drivers ▷ Operating Systems ▷ Utilities	Versio Info
d or remove SDK software compo perating Systems Drivers CMS Drivers Cype to Titler Name Caset Ca	IS Drivers Utilities Middleware Board Components A Description ADC Driver Clock Driver Clock Driver COMMON Driver CTime Driver DAC Driver GPID Driver IC Driver IC Driver IC Driver IC Driver	bstraction Layer Software Component		type to filter Name Description ⇒ Borvers ⇒ Boperating Systems ⇒ Butlitites	Versio Info
d or remove SDK software compo perating Systems Drivers CMS Drivers (syste to filter Name Composition	nents IS Drivers   Utilities   Middleware   Board Components   A Description ADC Driver CAPT Driver Clock Driver COMMON Driver CTMer Driver GPIO Driver I2D Driver I2D Driver I4D Driver I4	bstraction Layer Software Component ✓ Version 2.5.0 2.10 2.3.1 2.2.1 2.2.1 2.0.2 2.1.0 2.0.2 2.1.0 2.0.2 2.1.0 2.0.2 2.1.0 2.0.1		type to filter  tame Description  Descriptio	Versio
d or remove SDK software compo perating Systems Drivers CMS Drivers (syste to filter Name C adc C cont C cock C co	IS Drivers Utilities Middleware Board Components A Description ADC Driver CAPT Driver Clock Driver COMMON Driver CTime Driver ORC Driver BPIC Driver 12D Driver 12D Driver 12D Driver 12D Driver 12D Driver 12D Driver	bstraction Layer Software Component Version 2.5.0 2.1.0 2.3.1 2.3.1 2.2.1 2.0.2 2.1.0 2.0.2 2.1.0 2.0.2 2.1.0 2.0.2 2.1.0 2.0.2 2.1.0 2.0.2 2.1.0 2.0.2 2.1.0 2.0.1 2.1.0 2.0.2 2.1.0 2.0.1 2.1.0 2.0.1 2.1.0 2.0.1 2.0.2 2.0.1 2.0.1 2.0.1 2.0.2 2.0.1 2.0.2 2.0.1 2.0.1 2.0.2 2.0.1 2.0.2 2.0.1 2.0.2 2.0.1 2.0.2 2.0.1 2.0.2 2.0.2 2.0.2 2.0.2 2.0.1 2.0.2 2.0.2 2.0.2 2.0.1 2.0.2 2.0.2 2.0.2 2.0.1 2.0.2 2.0.1 2.0.2 2.0.1 2.0.2 2.0.1 2.0.1 2.0.5 2.0.1 2.0.5		type to filter lame Description ≥ Dorivers ≥ Coperating Systems ≥ Utilities	Versio Info
d or remove SDK software compo perating Systems Drivers CMS Drivers type to filter Name Composition Co	nents IS Drivers   Utilities   Middleware   Board Components   A Description ADC Driver CAPT Driver COMMON Driver CTimer Driver COMMON Driver CTimer Driver GPIO Driver I20 Driv	bstraction Layer Software Component ✓ Version 2.5.0 2.1.0 2.3.1 2.2.1 2.0.2 2.1.7 2.0.0 2.0.1 2.0.1 2.1.0 2.0.1 2.1.0 2.1.0 2.1.0 2.1.0 2.1.0 2.1.0 2.1.1 2.1.0 2.1.0 2.1.1 2.1.0 2.1.0 2.1.1 2.1.0 2.1.1 2.1.0 2.1.1 2.1.0 2.1.1 2.1.0 2.1.1 2.1.0 2.1.1 2.1.1 2.1.0 2.1.1 2.		type to filter Name Description > ≣ Drivers > ≣ Operating Systems > ≣ Utilities	Versio Info

- 3. Add the *OLED* library and configure the display operation as in the previous manual.
- 4. In *Config Tools -> Clocks*, change the frequency of the *FRO\_OSC* generator to 30 MHz.
- 5. Go to the main project file and modify the code as below:

```
#include <stdio.h>
#include "board.h"
#include "peripherals.h"
#include "pin_mux.h"
#include "clock_config.h"
#include "LPC804.h"
#include "fsl_debug_console.h"
#include "oled.h"
 char sbuff[32];
 volatile uint16_t adcValue = 0;
  * @brief Application entry point.
 int main(void) {
                  /* Init board hardware. */
BOARD_InitBootPins();
BOARD_InitBootClocks();
                   BOARD InitBootPeripherals();
 #ifndef BOARD_INIT_DEBUG_CONSOLE_PERIPHERAL
                  /* Init FSL debug console.
BOARD_InitDebugConsole();
 #endif
                        Initialize OLED *
                  OLED_Init(I2C0_PERIPHERAL);
                  while(1) {
                                    OLED_Clear_Screen(0);
sprintf(sbuff, "ADC: %5d", adcValue);
OLED_Puts(0, 1, sbuff);
OLED_Refresh_Gram();
                  }
                   return 0 ;
 1
```

6. Connect the display and check its operation.



#### II. A/D Converter

1. Go to Config Tool -> Pins and create a new preset called BOARD\_InitADCPins:



2. Click on the ADC block and connect the *ADC0* signal (*PIO0\_1* pin). Disable the default *Pull-Up* by setting the *Mode* field to Inactive:

3. Go to Clocks and turn on the ADC clock 5 MHz for the A / D converter:



4. Go to ADC settings and enter the following configuration:



5. Go to Peripherals, select *CTIMER* and configure it to change the state of the output at a frequency of 20 Hz:

• 📰 🐨 🔛 ZPSW_Lab07	M 🔒	Update Code • Functional Group	BOARD_InitPeripherals 💽 🔽 🐨 : 🥙 : 🖤 🖬 : 🖤 : 📰 🚺 : 🕲 : 🕼 : 🗇 : 🕼 : 🖓 : 🖓 : 🥙 🛷 • : 🎜 :	[] - (j) - 약 다 다 다 · ·		A : ■ × ■ ₪ § Q
Components 🕄 🦞 Peripherals		🚹 I2CO 🔮 CTIMERO 😫			A Overview 😫 🛃 Code Preview	
	0 1	Standard counter/timer (P	nipheral drivers (Device specific)]	🐪 🜑	> Configuration - General Info	
		Name CTIMEDO		Custom name	V Configuration - HW Info	
Peripheral drivers (Device specific)	0	CTIMERO		Custom name	Processor: LPC804	
	1200	Mode Input Capture/Match	Peripheral CTIMER0		Part number: LPC804M101JDH24	
		✓ <sup>6</sup> Timer/counter general configurat	ion	Preset Custom 🔻	Core: Cortex-M0P	
Custem initialization	0	<ul> <li>Timer counter configuration</li> </ul>			Board: LPCXpresso804	
		Timer mode	Timer (bus clock source)	¥	SDK Version: ksdk2_0	
		Bus clock source	System clock - BOARD, BootClockERO18M: 15 MHz, BOARD, BootClockERO24M: 12 MHz, BOARD, BootClockERO30M: 15 MHz		> Project	
		Clock source frequency	15 Milit (BOARD, BootClockERO18M)		M. Bulabarata	
		Times issued frequency developmentales			Periprierais	a sha diffui a sala basal dalama
		Calculated prescaler	1		Compares the intanzation o	sine son periprisi al drivers.
		Calculated timer input frequency	15 MHz; 66.867 ns		(Ψ)	
		Timer counter period	50 ms			
		Start timer in initialization code	2			
					✓ Generated code	
		✓ <sup>4</sup> Match channels +	x		Update code enabled	
		Match 3			board/peripherals.c	
		Changel ID	Match 2		board/peripherals.h	
		Match channel O	4 Mateh shaneel 2			
		- Match channel	Match channel 3		<ul> <li>Functional groups</li> </ul>	
		Channel frequency/period/offset	750000		🛃 80ARD_InitPeripherals	
		Calculated match frequency/period	Influet 20.Hz: 50 ms			
		Enable counter reset on match			<ul> <li>Other tools</li> </ul>	
		Enable counter stop on match			$\cap$	
		Output control	Topole bitloutout			
		Initial output value				
		Enable match interrupt request				
		and the second sec				
		capture channels     +				
		Add item by clicking on plus button				
		<ul> <li>Interrupt and callback settings</li> </ul>				
		Configuration disabled (any m	atch/capture/pwm channel interrupt is not enabled)		A Problems 🖾	BY
		V Interrupt				
		Interrupt	TIMERO IRON		Level v Resource	Issue
		Enable priority initialization			S Warning CTIMER0.ctimer_match.3	Match output of the match 3
		Priority			S Warning USART0	Peripheral USARTO is not ini
		Callback mode Callbar	ck disabled			

The *ADC* converter will be triggered by only one edge, therefore its sampling frequency will be twice lower - i.e. 10 Hz.

6. Go to the main project file and modify the code as below:



7. Connect the potentiometer to the board, program the microcontroller and check the example. By moving the potentiometer axis, the displayed value should change in the range of 0-4095 (12-bit resolution), which corresponds to the input voltage of 0-3.3 V.



#### III. GUI - a simple analog indicator

1. Modify the project code:

```
#include <stdio.h>
#include "peripherals.h"
#include "peripherals.h"
#include "pin_mux.h"
#include "lock_config.h"
#include "fsl_debug_console.h"
#include "fsl_debug_console.h"
#include "fsl_debug_console.h"
#include "fsl_debug_console.h"
#include "fsl_debug_console.h"
#include "fsl_debug_console.h"
#include "lock_n"
#include "lock_n"
#include "lock_n"
#include "lock_n"
#include "sl_debug_console.h"
#includes the sl_d
```

# Programing of embedded systems

7. A/D Converter

/* ADC_SE	<pre>iQA_IRQn interrupt handler */ ADC_SEG_A TRANSHEE(width) {</pre>
VOID ADC_	/* Get status flags */
	<pre>if (kADC_convSeqAInterruptFlag == (kADC_convSeqAInterruptFlag &amp; ADC_GetStatusFlags(ADC_PERIPHERAL))) {</pre>
	/* Place your interrupt code here */
	ADC_GetChannelConversionResult(ADC_PERIPHERAL, 0, gAdcResultInfoPtr);
	accvalue = gAccresuitInfoStruct.result;
	ADC ClearStatusFlags (ADC PERIPHERAL, KADC ConvSegAInterruptFlag):
	}
}	
/*	
* @brief	Application entry point.
<pre>int main(</pre>	void) {
#ifndef E #endif	<pre>/* Power on ADC. */ POWER_DisablePD(kPDRUNCFG_PD_ADC0); /* Init board hardware. */ BOARD_InitBootPins(); BOARD_InitBootClocks(); BOARD_InitBootPeripherals(); 30ARD_INIT_DEBUG_CONSOLE_PERIPHERAL /* Init FSL debug console. */ BOARD_InitDebugConsole(); /* Initialize OLED */ OLED_Init(I2C0_PERIPHERAL); </pre>
	while(1) {
	<pre>0LED_Clear_Screen(0);</pre>
	<pre>data=adcValue/4095.0;</pre>
	Gauge(64, 32, 32, data); sprintf(sbuff, "%3d‰", (uint8_t)(data∗100)); OLED_Puts(50, 7, sbuff);
	OLED_Refresh_Gram();
	return 0 ;
r	

2. Build the project in **Release** mode, program the microcontroller and check the example.

#### **IV. Exercises**

- 3. Modify the appearance of the analog indicator as you see fit.
- 4. Write a function that draws the *n-last* samples in the form of a bar graph. The graph is to move across the display screen (horizontally or vertically).