# LABORATORY MANUAL

# 18CSL48 - MICROCONTROLLER AND EMBEDDED SYSTEMS LABORATORY

2019-20



DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING ATRIA INSTITUTE OF TECHNOLOGY Adjacent to Bangalore Baptist Hospital Hebbal, Bengaluru-560024

# **SYLLABUS**

#### MICROCONTROLLER AND EMBEDDED SYSTEMS LABORATORY

#### **SEMESTER - IV**

# Subject Code 18CSL48 Exam Marks 60

IA Marks 40 Exam Hours 03

Course Learning Objectives: This course (18CSL48) will enable students to:

Develop and test Program using ARM7TDMI/LPC2148

• Conduct the experiments on an ARM7TDMI/LPC2148 evaluation board using evaluation version of Embedded 'C' & Keil Uvision-4 tool/compiler.

**Programs List:** 

#### PART A

Conduct the following experiments by writing program using ARM7TDMI/LPC2148 using an evaluation board/simulator and the required software tool.

1. Write a program to multiply two 16 bit binary numbers.

2. Write a program to find the sum of first 10 integer numbers.

3. Write a program to find factorial of a number.

4. Write a program to add an array of 16 bit numbers and store the 32 bit result in internal RAM

5. Write a program to find the square of a number (1 to 10) using look-up table.

6. Write a program to find the largest/smallest number in an array of 32 numbers .

7. Write a program to arrange a series of 32 bit numbers in ascending/descending order.

8. Write a program to count the number of ones and zeros in two consecutive memory locations.

#### PART –B

Conduct the following experiments on an ARM7TDMI/LPC2148 evaluation board using evaluation version of Embedded 'C' & Keil Uvision-4 tool/compiler.

9. Display "Hello World" message using Internal UART.

10. Interface and Control a DC Motor.

11. Interface a Stepper motor and rotate it in clockwise and anti-clockwise direction.

12. Determine Digital output for a given Analog input using Internal ADC of ARM controller.

13. Interface a DAC and generate Triangular and Square waveforms.

14. Interface a 4x4 keyboard and display the key code on an LCD.

15. Demonstrate the use of an external interrupt to toggle an LED On/Off.

16. Display the Hex digits 0 to F on a 7-segment LED interface, with an appropriate delay in between

# Laboratory Outcomes: The student should be able to:

• Develop and test program using ARM7TDMI/LPC2148

• Conduct the following experiments on an ARM7TDMI/LPC2148 evaluation board using evaluation version of Embedded 'C' & Keil Uvision-4 tool/compiler.

## **Conduct of Practical Examination:**

• Experiment distribution o For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity. o For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.

• Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.

• Marks Distribution (Courseed to change in accoradance with university regulations) g) For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+70+15 = 100Marks h) For laboratories having PART A and PART B i. Part A – Procedure + Execution + Viva = 6 + 28 + 6 = 40 Marks ii. Part B – Procedure + Execution + Viva = 9 + 42 + 9 = 60Marks

# <u>LPC2148\_ARM 7 Development Board</u>



# Introduction

Thank you for buying **ARM 7 development KIT** *based* on Philips ARM7TDMI LPC2148 microcontroller. This document is a User's Guide that describes the **ARM 7 (LPC2148) development KIT** hardware design and some basic software interface principles with the Hardware

**LPC2148 ARM 7 Laboratory system** is a methodology to impart the micro processor concepts through a practical approach. It aims at bridging the gap between the theoretical knowledge and the practical scenario. In the process, the entire structure has been broken down to simpler concepts and takes a guided approach to the complex concepts and development process.



# **ARM 7 Development Kit Features**

**ARM7** (**LPC2148**) **STARTER-KIT** with Philips ARM7TDMI LPC2148 microcontroller lets you get upand-running quickly. The small form factor board offers many unique features that ease your learning curve and program development.

#### ARM LPC2148

- 16/32-bit ARM7TDMI-S microcontroller in a tiny LQFP64 package.
- 40 kB of on-chip static RAM and 512 kB of on-chip flash program memory.
- USB 2.0 Full Speed compliant Device Controller with 2 kB of endpoint RAM.

- In addition, 8 kB of on-chip RAM accessible to USB by DMA.
- Two 10-bit A/D, with conversion times as low as 2.44 µs per channel.
- Single 10-bit D/A converter provide variable analog output.
- Two 32-bit timers/external event counters and watchdog.
- Low power real-time clock with independent power and dedicated 32 kHz clock input.
- Multiple serial interfaces including two UARTs (16C550), two Fast I2C-bus (400 kbit/s), SPI and SSP with buffering and variable data length capabilities.
- Vectored interrupt controller with configurable priorities and vector addresses.
- 60 MHz maximum CPU clock available from programmable on-chip PLL with settling time of 100 μs.
- Up to 45 of 5 V tolerant fast general purpose I/O pins in a tiny LQFP64 package.
- 2 x 16 Character LCD Display.
- Standard JTAG connector with ARM 2x10 pin layout for programming/debugging with
- H-JTAG
- USB connector, USB link LED
- RS232 Connector.
- LM35 temperature sensor connected to ADC
- On board voltage regulator 3.3V with up to 800mA current.
- Single power supply: 12V DC.
- Power supply LED.
- Power supply filtering capacitor.
- RESET button
- 12 MHz crystal.

# **ARM 7 Development board components and features**

#### **1 POWER SUPPLY**

The way of applying power to the ARM7 STARTER- KIT:



• First method applies DC power supply (+12V) from the lab power supply to **J1** jack

#### 2 CPU

The NXP LPC2148 provided with the Starter- KIT is a high-end device with advanced ADC and DAC features, USB 2.0 Full Speed interface, PWM features, have SPI, SSP and two I2C interfaces. A 12 MHz crystal provides the clock signal to the microcontroller. This crystal is used for develop USB applications. The maximum execution speed is 12.000\*5 = 60 MHz.

The crystal frequency may be changed to any desired value. The board also contains a 32.768 kHz crystal that is used by the on-chip real-time clock, and +3V backup battery for save time register values in power down mode.

#### **3 USB COMMUNICATION**



LPC2148 Starter- KIT support USB 2.0 12 Mbit Full Speed function. USB D+ and D- are connected to CP2102 USB Bridge driver which allows Downloading hex file directly from USB interface. The **LED 9** yellow led will fire, when device configured and connected to the USB Bus. Standard USB B connector used for connect board to the PC Host.

#### **4 SERIAL COMMUNICATION**

On board RS232 serial port is connected to the UART1 of the microcontroller, which may be used to interface the target ARM board with host PC/any serial terminal for serial data acquisition? Level transition buffer MAX3232 has been used to make RS232 TTL compatible.



#### 5. LCD Display

DB7	P0.13
DB6	P0.12
DB5	P0.11
DB4	P0.10
DB3	P0.09
DB2	P0.08
DB1	P0.07
DB0	P0.06
E	P0.28
R/W	P0.29
RS	P0.22

#### **6 STEPPER MOTOR INTERFACE**

Connect the Stepper motor which is supplied with the kit to CN5



#### 7. LED INTERFACE

8 LED s are connected to P1.16 to P1.23



#### 8.7 SEGMENT DISPLAY



**9. RELAY INTERFACE** Relay control pin connected to P1.25



#### **10. BUZZER INTERFACE**

Buzzer control pin is connected to P1.24



#### **11. EXTERNAL INTERRUPT SWITCHES**

There are two external interrupt switches connected to P0.15 and P0.20



#### 12. GENERAL PURPOSE PINOUT AND GND PINOUT

There are 8 general purpose pin outs connected to P1.16 to P1.23 and GND for ground reference. And 10 pin FRC connector connected to pin no P1.24 to P1.31



#### **13. RESET SWITCH**

This is used to reset the excecutable program.



#### 14. ADC block

LM35 temperature sensor is directly connected to AD0.3 port P0.30.



#### 15. DAC block

Analog Out pin is connected to P0.25. Here you can see the generated waveform using CRO probe connected to TP5 test point and Gnd pin.



### 16. 4x4 matrix keypad

16 keys are connected as follows



## **17. DC motor interface block**

Connect supplied DC motor to J4 connector

This DC motor controller block is equipped with L293D H bridge driver for Both speed and direction control

PWM5 is connected to enable pin P0.21. Different PWM signal will control the speed.

TMS=P1.30, TRST=P1.31. For motor rotation any one pin should be High and at the same time another should be Low. Make it Vice versa for reverse rotation



#### 18. General purpose switch

This is connected to P0.23



You can use this switch for any general purpose like Stepper /DC motor direction change, Toggling LED etc.

# **Creating Project in KEIL and Generating HEX file**

The UE-ARM7-SK kit is compatible with many commercial and open

source tools. The kit comes along with a evaluation version of KEIL IDE with 32K Code limit. This limit is sufficient enough for the experiments to be conducted. The installer for the same is provided in the CD that comes along with the kit. Follow the simple steps as requested by the installer. The following section explains how to create a KEIL project for the experiments. We will create a example project which will blink the LEDs on the kit. The section assumes that you already have the necessary '.c' and '.h' files required for the project in a specified folder

# Run the KEIL IDE and go the

👿 D:WLLAS_201	011\ullas_arm\blinkled_easy\led1.uvproj - µVision4	
File Edit View	Project Flash Debug Peripherals Tools SVCS Window Help	
🗋 💕 🛃 🕼	New µVision Project	
🕺 🕮 🕮 🥥 🗄	New Multi-Project Workspace	
Project 4	Open Project	
🖃 🔁 Target 1	Close Project	
🗄 📇 Source Gr	Export	•
	P Manage	+ Functi
Startu	Select Device for Target 'Target 1'	

2. Lets give the project a name and save it as shown below

Create New Pr	oject				? 🔀
Save in:	Blink_led		•	🗢 🔁 💣 📰 •	
My Recent Documents Desktop My Documents My Computer	Blink_led.Uv2	2			
My Network	File name:	Blink_led.Uv2		•	Save
ridces	Save as type:	Project Files (*.uv2)		•	Cancel

3. Select the CPU target as LPC2148 under the option of NXP in the list

Select Device for Target 'Target 1'		$\mathbf{\times}$
CPU Vendor: NXP (founded by Philips) Device: LPC2148 Toolset: ARM		
Lata base	ARM7TDMI-S based high-performance 32-bit RISC Microcontroller with Th 512KB on-chip Flash ROM with In-System Programming (ISP) and In-Applix Two 10bit ADCs with 14 channels, USB 2.0 Full Speed Device Controller Two UARTs, one with full modem interface. Two I2C serial interfaces, Two SPI serial interfaces Two 32-bit timers, Watchdog Timer, PwM unit, Real Time Clock with optional battery backup, Brown out detect circuit General purpose I/O pins. CPU clock up to 60 MHz, On-chip crystal oscillator and On-chip PLL	
	OK Cancel Help	

4. Next click YES option in the pop-up window

VISION			2
Copy P	hilips LPC2100 Startup Co	de to Project Folder and	Add File to Project

- 5. Go to file and select new. In that edit the code and click on save.
- 6. Give the file name as blink\_led.c
- 7.Goto project in that click on option as shown below.

👿 C:Wocument	s and	Setti	ings\U	ILLAS\D	esktop\dfgd	\as.uvp	roj - µ	Vision4		
File Edit View	Proj	ect F	Flash	Debug	Peripherals	Tools	SVCS	Window	Help	
		New J	µVisior Multi-F	n Project Project We	 orkspace					
Project	<b>L</b>	Open	n Projec	rt						
🛨 🔁 Target 1		Expor	rt							•
		Mana	ge							• <sup>Fu</sup>
		Select	t Devic	e for Targ	et 'Target 1'					
		Remo	ove Iten	n						2 t
	**	Optio	ons for	Target 'Ta	arget 1'					Alt+F7
		Clean	n target	t						
		Build	target							F7 <sup>L.E</sup>
		Rebui	ild all t	arget file	s					,ED
1		Batch	n Build.							

8. In option click on output and mark create hex file as shown below. And go to linker make below changes

🖫 Options for Target 'Target 1'	×
Device Target Output Listing User C/C++ Asm Linker Debug	Utilities
Select Folder for Objects Name of Executable: as	
<ul> <li>Create Executable: .\as</li> <li>Debug Information</li> <li>Create HEX File</li> <li>Browse Information</li> <li>Create Library: .\as.LIB</li> </ul>	Create Batch File
OK Cancel D	refaults Help
V Options for Target 'Target 1'	
Device Target Output Listing User C/C++ Asm	Linker Debug Utilities
Use Memory Layout from Target Dialog	
Make RW Sections Position Independent Make RO Sections Position Independent	R/O Base: 0x0000000 R/W Base 0x40000000

Options for Target 'Target 1'			×
Device Target Output Listing	User C/C++ Asm	Linker Debug Utilities	
NXP (founded by Philips) LPC214 Operating system: None	8 <u>X</u> tal (MHz): 12.0	Code Generation ARM-Mode	<b>n</b> Bjg Endian
Read/Only Memory Areas		Read/Write Memory Areas	
default off-chip Start	Size Startup	default off-chip Start	Size Nolnit
ROM1:	C	RAM1:	
ROM2:	C	RAM2:	
ROM3:	o	RAM3:	
on-chip		on-chip	
IROM1: 0x0	0x80000 (•	IRAM1: 0x40000000	0x8000
IROM2:	0	IRAM2:	
	ок	ancel Defaults	Help

And also check Use MICRO LIB as follows

9. Add the file to source group as shown below.

🕼 C: D	ocur	nents and SettingsWLLASWesktop\dfgd\as.uvp	roj - µVision4
File Ed	lit	View Project Flash Debug Peripherals Tools	SVCS Window
🗋 🗁		● * 日間 • ・ ・ + + や 作 作 作	: *# *# //# //#
I I I I I I I I I I I I I I I I I I I		🧼 🔜 📫 Target 1 🔹 💰 📥 🗟	3
Project		A X Dinkled.c	
🖃 - 🔁 Ta	arget	1 01 #include <lpc214x.h> //</lpc214x.h>	/ LPC2148 MP
·····	*	Options for Group 'Source Group 1' Alt+F7	long int);
		Open File	
		Open List File	
		Open Map File	Set GPI00.
		Rebuild all target files	ue //
		Build target F7	Clear Pin
		Translate File	isplay LED
		Stop build	Set Pin PO isplay LED
		Add Group	
		Add Files to Group 'Source Group 1'	*/
		Remove Group 'Source Group 1' and its Files	*/
	♣	Manage Components	long int c
	$\checkmark$	Show Include File Dependencies	t1;} // L
	_	23 } 24	4

10. Select .C file as shown below

Add Files to	Group 'Source Group 1'				? ×
Look in: 🗀	dfgd	•	🗈 💣	:::: <b>-</b>	
MSG Root blinkled Date Size:	: C File Modified: 1/11/2012 12:42 PM 648 bytes				
File name:	blinkled			Add	
Files of type:	C Source file (*.c)		•	Close	
					1.

11. Go to project and click on build target.

File Edit	View	Proj	ect Flash	Debug	Peripherals	Tools	SVCS	Window	Help		-
i 🗋 💕 🔒	0		New µVisio	n Project							[
🔹 🔛 🔛			New Multi	Project W	orkspace						
Project	ŗ		Open Proje	ect							
E Target	• 1		Close Proje	ct							
E G So	urce Gr		Export							•	
			Manage							•	(ncti)
			Select Devi	ce for Targ	et 'Target 1'						
			Remove Ite	m							
		ĸ	Options fo	r Target 'Ta	arget 1'					Alt+F7	
			Clean targe	et							1
			Build targe	t						F7	(סי
			Rebuild all	target file	s						9
			Batch Build	i							
		٢	Translate C	:\Documer	nts and Setting	gs\ULLAS	\Deskto	p\dfgd\blin	kled.c	Ctrl+F7	

If code is correct then it shows like

Build Output Program Size: Code=584 RO-data=16 RW-data=0 ZI-data=96 FromELF: creating hex file... "as.axf" - 0 Error(s), 0 Warning(s). Build target files

The hex file of the code will be generated now.

# Working with Flash Magic Software

Click the start button, select the programs options and select the flash magic software. The main screen appears

≥ 🗔   Q, 🗃 🧔	8 🗸 🎩 🔊 🛛	A 😮 😂		
Step 1 - Communicati	ons	Step 2 - Erase		
COM Port:	COM 3	Erase block 0 (0x0	00000-0x000FFF)	^
Baud Rate:	9600 💌	Erase block 2 (0x0	02000-0x002FFF)	-
Device:	LPC2148 -	Erase block 3 (0x0 Erase block 4 (0x0	03000-0x003FFF) 04000-0x004FFF)	
Interface:	None (ISP) -	Erase block 5 (0x0	05000-0x005FFF1	~
Oscillator Freq. (MHz):	12.000000	Erase all Flash+	ed by Hex File	
Step 3 - Hex File Hex File: C:\Docume	ents and Settings\demo1\	Desktop\ARM_demo\4	ARM H Brows	e
Step 3 - Hex File Hex File: C:\Docume Modified: To Step 4 - Options	ents and Settings\demo1\2 uesday, June 14, 2011, 2:2	Desktop\ARM_demo\v 88,42 AM g Step 5 - S	ARM H Brows	e Cilck
Step 3 - Hex File Hex File: C:\Docume Modified: To Step 4 - Options Verify after program	mis and Settings\demo1\0 uesday, June 14, 2011, 2:2 uming ┌╴ Set Code Read	Desktop\ARM_demo\4 8:42 AM g Step 5 - S	ARM H Brows tore info tart Start	e Click
Step 3 - Hex File Hex File: C \Docume Modified: Tu Step 4 - Options Verily after program Fill unused Flash Gan Diock, checka Execute	an ts and Settings\demo1\0 uesday, June 14, 2011, 2:2 uming ┌─ Set Code Read	Desktop\ARM_demo\4 8:42 AM g Step 5 - S I Prot	ARM H Brows tore info tartt Start	e <u>Cick</u>
Step 3 - Hex File Hex File: C \Docume Modified: To Step 4 - Options Verily after program Fill unused Flash Gen block checks Execute Visit the "Flash Magic www.esacademy.com	mis and Settings\demo1\2 uesday, June 14, 2011, 2.2 uming C Set Code Read ums " home page for info on the vooftware/flashmagic	Desktop\ARM_demo\4 8:42 AM g Step 5 - S I Prot e latest revision	ARM H Brows tore info tartt Start	e <u>Click</u>

Go to options in the main menu and select advanced options the following screen displays.

ep 1 - Communicatio COM Port	Сом 3 🗸	Step 2 - Erase Erase block 0 (0x000000-0	×000FFF)
dvanced Option	IS	12 11 1 4 10 001 000 0	
T1: 50 T1:	asserted while COM Port	open	
Assert DTR an			
Assett DTH an		Cancel	DK
Gen block checksur Execute		Cancel	DK

Select hardware config tab and uncheck the use DTR and RTS option and press the OK button

Select the comm. Port (Check in device manager about UART bridge COM port). Select the baud rate as 9600, select device as LPC2148, interface is NoneISP, Oscillator is 12.000

Browse the appropriate hex file provided on the CD and click on the Start Button. The program will the downloaded in the Kit.

#### **UART Program**

Hello word will be displayed in terminal

Open terminal in flash magic, select baud rate and com port. Press OK.

	Reash Magic - NON PRODUCTION USE ONLY	
	File ISP Options Tools Help	
🏟 Flash Magic - NON PRODUCTION USE ONI	Step 1 - Communications Step 2 - Erase	
File ISP Options Tools Help	Se Terminal Settings	
🛅 🗟 🔍 🧊 🍏 🖬 Terminal 📃	Port and Speed	
Step 1 - Communication Scripts	CUM Port: CUM3   Baud Rate: Sb00	
Select Device LPC2148		

# **DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING**

#### Vision:

To be centre of excellence in Information Science and Engineering equipping students with top notch competencies in the domain of information technology.

#### **Mission:**

- Promote best teaching learning , research, innovation and also instill professional ethics, cultural values and environmental awareness among the students
- Establishing learning ambience with best infrastructure facilities