MINI
DEVELOPMENT
BOARD

LPC
2148

USERS GUIDE

make your own
intelligent embedded world . . .

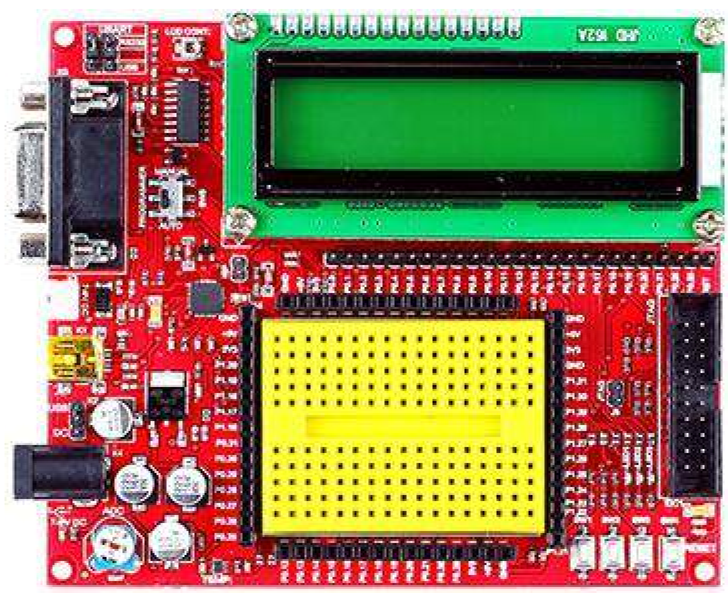
rhydoLABZ™

TABLE OF CONTENTS

1.0	**OVERVIEW**		2
	1.1	KEY FEATURES	3
	1.2	CONTROLLER SPECIFICATION	4
2.0	**HARDWARE INTRODUCTION**		6
	2.1	BLOCK DIAGRAM	6
	2.2	INTERFACE OVERVIEW	7
	2.3	PERIPHERAL DESCRIPTION	8
	2.4	JUMPER SET DESCRIPTION	9
	2.5	POWER SUPPLY	10
	2.6	CLOCK SOURCE	10
	2.7	MICROCONTROLLER - PIN OUT	11
	2.8	PORT EXPANDER	12
	2.9	JTAG CONNECTION	13
	2.10	LED INTERFACING	14
	2.11	LCD INTERFACING	15
	2.12	PULL-UP KEYS INTERFACING	16
	2.13	BUZZER INTERFACING	17
	2.14	UNIVERSAL ASYNCHRONOUS RECIEVER TRANSMITTER	18
	2.15	SERVO MOTOR	25
	2.16	ANALOG TO DIGITAL CONVERTER MODULE	25
	2.17	TEMPERATURE SENSOR INTERFACING	26
3.0	**SOFTWARE DEVELOPMENT**		27
	3.1	TOOLS AND SOFTWARE	27
4.0	**I/O PIN DISTRIBUTION**		41

OVERVIEW

ARM LPC2148 Mini Development Board is a miniature and powerful hardware platform to evaluate LPC2148 Flash memory microcontroller. The eCee ARM LPC2148 Board contains all hardware components that are required in a single-chip LPC2148 controller system plus 1 COM port for serial RS232 interface.



1.1. KEY FEATURES OF PIC16F877A DEVELOPMENT BOARD-MINI

1. Compact design and user friendly
2. LPC 2148 is integrated onto the board
3. No separate Programmer required (Built in Bootloader)
4. On-board 12 MHz crystal for controller
5. Multiple power input options (USB, RMC Connector, DC barrel jack) with jumper selection
6. Reverse supply voltage polarity protection
7. On-board 5V and 3.3V regulators
8. 3.3 and 5V output available on berg strip
9. Power indication LED(Red)
10. Servo, LCD & Zigbee can be easily interfaced through on-board connectors
11. Zigbee communication through UART0
12. Provision for Zigbee connectors on either side of the board
13. Potentiometer for LCD contrast control
14. On-board buzzer
15. 4 on-board switches including a reset switch & 3 others connected to port pins via jumpers
16. 3 on-board SMD LED s connected to port pins via jumpers
17. Potentiometer connected to ADC
18. Temperature sensor (MCP 9700)
19. On Board ICD Connector for Debugging
20. All port pins are accessible through both male & female berg strips
21. UART0 communication possible through Zigbee, DB9 connector and on-board connector
22. UART0 on can be used in 3.3V and 5V levels(with jumper selection)
23. UART1 communication possible through berg strip connector
24. Internal USB is accessible through on-board USB port
25. Breadboard can be attached to the board
26. 32.768 kHz crystal for internal RTC
27. Battery holder for external battery used to power RTC
28. Multiple programming options – USB, Serial port
29. Programmer switch to select 2 programming modes :
    - Auto - no reset, no ISP jumper
    - Manual - Press reset switch, use ISP jumper
30. ISP jumper should be removed for code execution
31. For UART0 communication, programmer switch should be in manual mode
32. Professional EMI/RFI Complaint PCB Layout Design for Noise Reduction
33. High quality two layer PTH PCB

1.2. CONTROLLER SPECIFICATION

1. High Performance 32-bit ARM7TDMI-S™ CPU
2. 512 kB Programmable Flash Memory provides minimum of 10,000 erase/write cycles and 10 years of data-retention.
3. 32 kB + 8 kB Data Memory (SRAM)
4. Provides 8 kB of on-chip RAM accessible to USB by DMA
5. Two 10-bit ADCs provide a total of 14 analog inputs, with conversion times as low as 2.44 us per channel
6. Single 10-bit DAC provides variable analog output
7. Two 32-bit Timers/External event counters
8. Four Capture and four Compare channels
9. PWM unit with six output pins
10. Low power Real-time clock with independent power and dedicated 32 kHz clock input
11. Multiple serial interfaces including two UARTs (UART0 & UART1), two Fast I2C (400 kbit/s), SPI™ and SSP with buffering and variable data length capabilities
12. Vectored interrupt controller with configurable priorities and vector addresses
13. Up to 45 of 5 V tolerant general purpose I/O pins
14. Up to nine edge or level sensitive external interrupt pins
15. USB 2.0 Full-speed compliant device controller with 2 KB of endpoint RAM
16. In-System Programming/In-Application Programming (ISP/IAP) via on-chip boot loader software
17. Single Flash sector or full chip erase in 400 ms and programming of 256 bytes in 1 ms.
18. Embedded ICE RT and Embedded Trace interfaces offer real-time debugging with the on-chip Real Monitor software and high-speed tracing of instruction execution
19. Port pins P1.0-P1.15 & P0.24 are not externally accessible. P0.31 is output only digital pin. P0.26 & P0.27 cannot be used as GPIO pins since these are dedicated to USB (D+ & D- respectively)
20. 60 MHz maximum CPU clock available from programmable on-chip Phase-Locked Loop (PLL) with settling time of 100us
21. On-chip integrated oscillator operates with external crystal in range of 1 MHz to 25 MHz
22. Power saving modes include Idle and Power-down
23. Individual enable/disable of peripheral functions
24. Processor wake-up from Power-down mode via external interrupt
25. Single power supply chip with Power-On Reset (POR) and Brown-Out Detection (BOD)
26. CPU operates in the range of 3-3.6 V

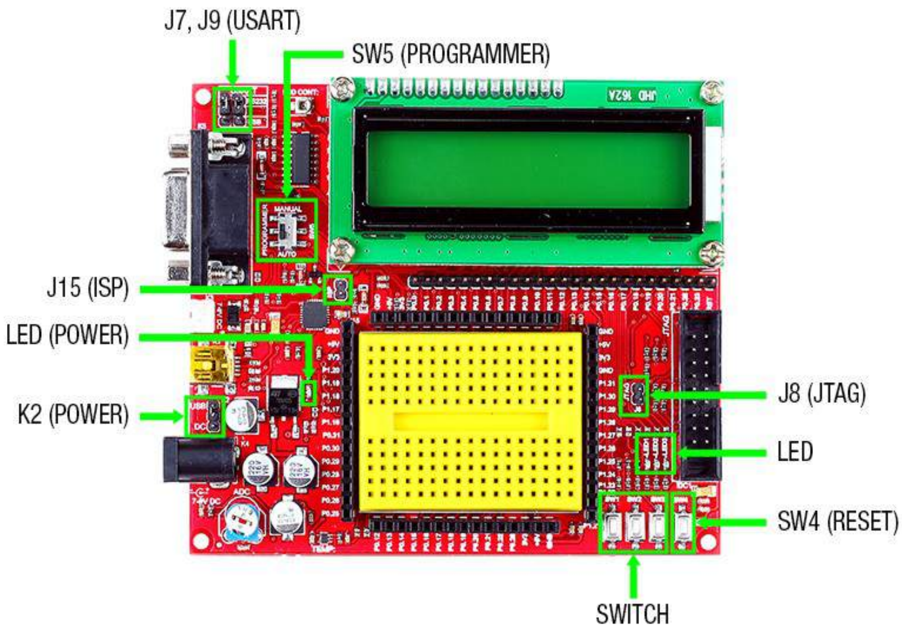
PACKAGE CONTENTS

- Fully Assembled and Tested eCee ARM LPC2148 Mini Development Board
- Software CDROM with
  - Schematic
  - Programming Software
  - Sample Hex Code
  - Example Codes

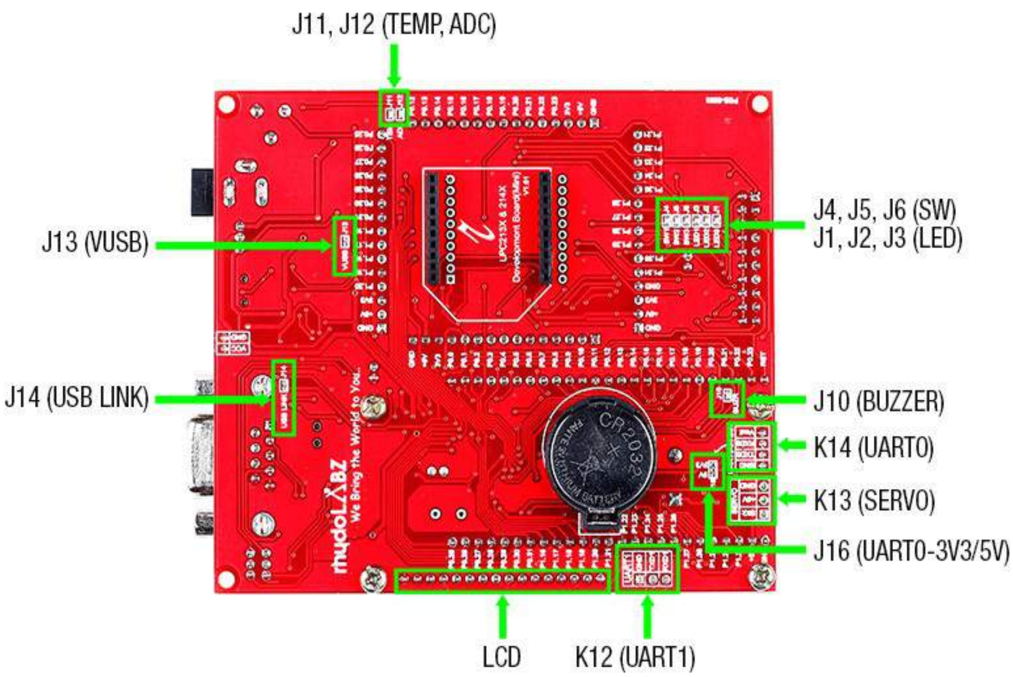
HARDWARE INTRODUCTION

2.1. BLOCK DIAGRAM

2.2. INTERFACE OVERVIEW



Top view of the Development board -Mini



Bottom view of the Development board -Mini

2.3. PERIPHERAL DESCRIPTION

PERIPHERALS	DESCRIPTION
ISP(J15)	ISP Connector,To program the IC
K6,K7	PORT pins on male berg strip
K1	USB Socket
K2	To select power source as USB/DC
K3	2 pin male RMC connector for power
K4	DC barrel jack
K5	UART Interface via Female DB9 Connector
K14	RMC connector for 3V3/5V UART0
K13	Servo connector pin
K8-K9-K10-K11	PORT pins on female berg strip
LED1-LED3	Light Emitting Diodes
PWR	Power indication LED
SW1-SW3	Pull-Up Switches
RESET(SW4)	Reset Button
PROGRAMMER(SW5)	To select Auto/ Manual mode of programming
LCD CONT (P1)	LCD Contrast control Pot
ADC (P2)	Potentiometer used as ADC input
U1	ARM LPC 2148
U2	LM7805(5V regulator IC)
U3	LD1117 (3V3 regulator IC)
U4	CP2102(USB interface)
U5	MAX232(Level converter)
U6	Temperature Sensor (MCP9700)
U7	Zigbee module connectors
IDC1	JTAG connector
BUZ1	Buzzer
LCD1	LCD

2.4. JUMPER SET DESCRIPTION

JUMPER No.	DESCRIPTIONS	SET OPTIONS	SETTINGS DESCRIPTION
K2	Power Supply Options	1-2	Select USB power
		2-3	Select external DC power
J16	UART0	Short to access	Select 3.3/5 V level for UART communication via RMC connector
J12	Potentiometer	Short access	Enables ADC connection via POT
J11	Temperature Sensor	Short access	Enables temp sensor connection
J1, J2, J3	LED	Short access	Enables LED connection
J4,J5,J6	Pull-Up Key	Short access	Enables Pull-Up Key connection
J10	Buzzer	Short access	Enables buzzer connection
J8	JTAG	Short access	Establish J connection
J7, J9	USART	1-2	RS232 Connection
		2-3	USB Connection

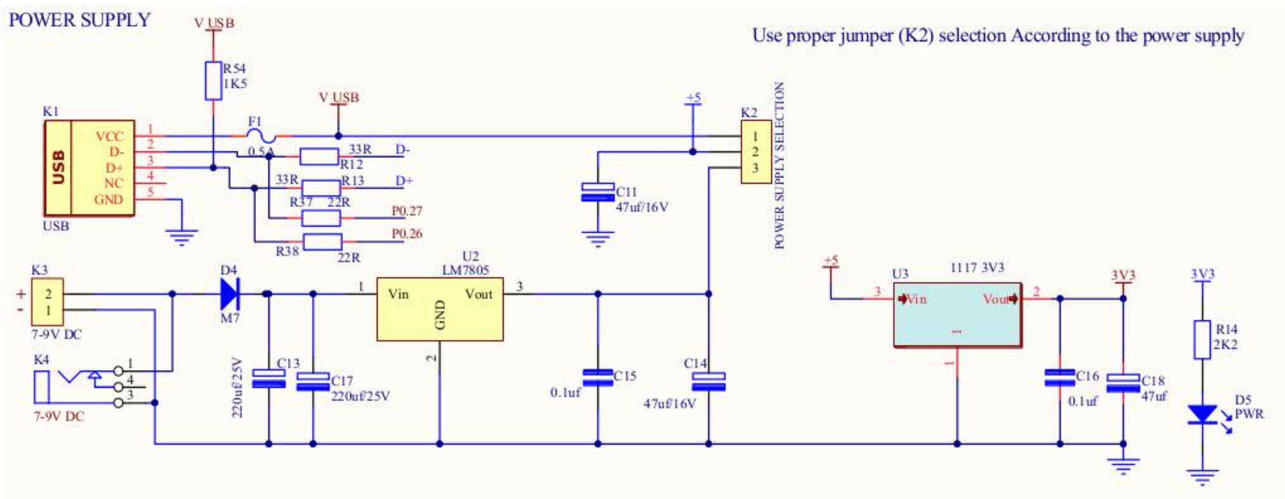
2.5. POWER SUPPLY

ARM LPC2148 Mini Development Board has 3 provisions for giving power supply input

- USB connector
- DC Barrel Jack Connector
- 2 Pin Male RMC Connector

The input source can be selected as DC/USB using the jumper. If DC source is selected, then either DC Barrel Jack or RMC connector can be used and the supply voltage should be in the range of 7-12 V. Once the board is powered, the power LED(red LED on the board) glows.

The external Power Supply circuit is given below:

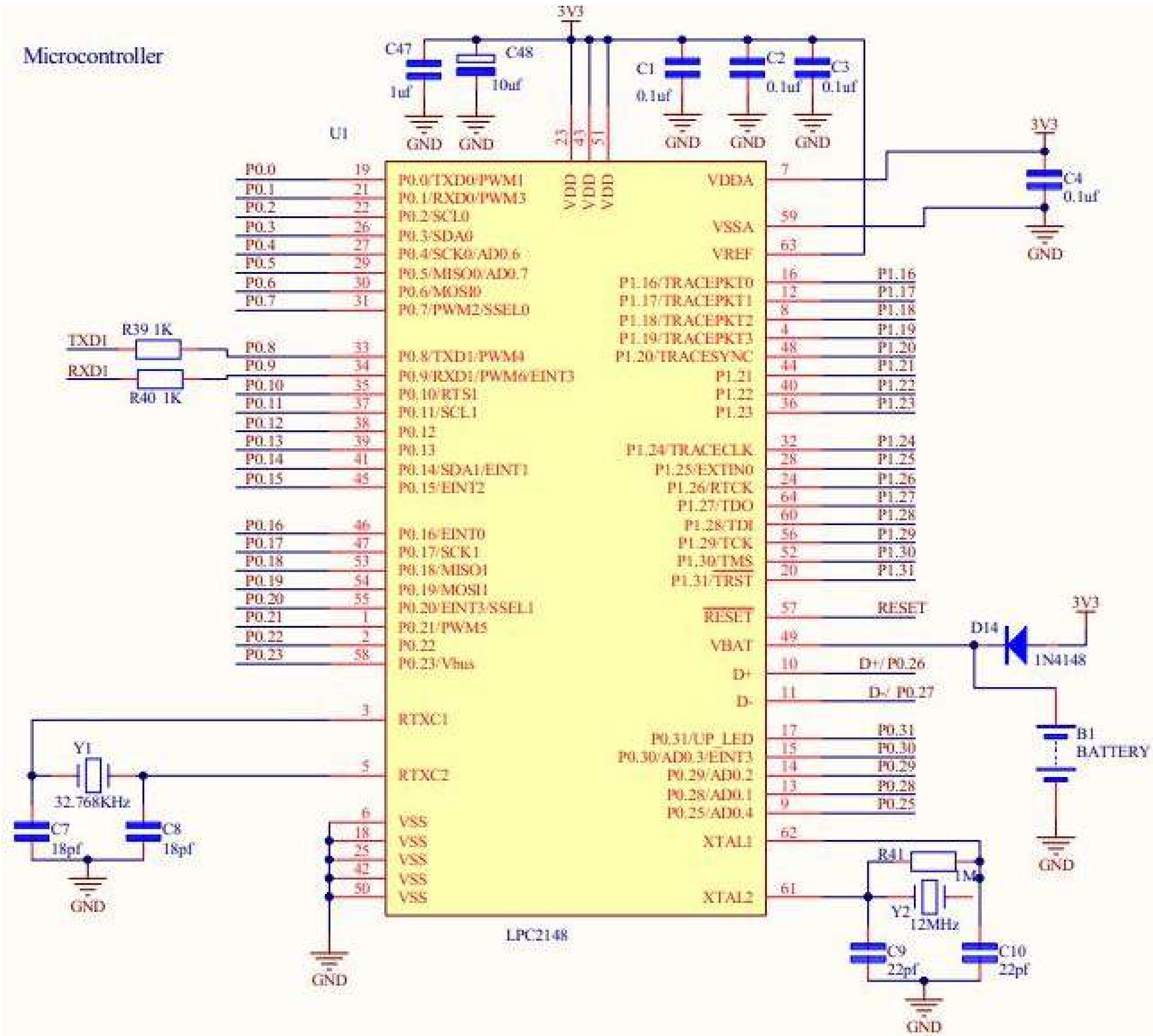


2.6. CLOCK SOURCE

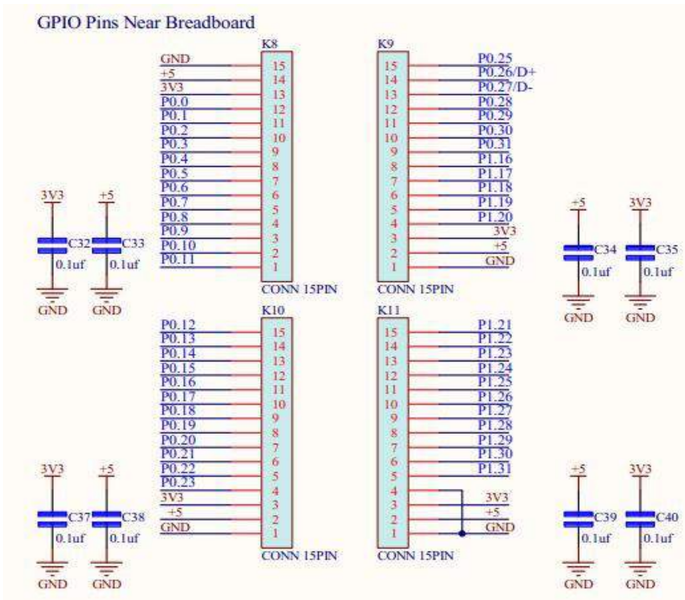
LPC2148 Mini Development Board uses

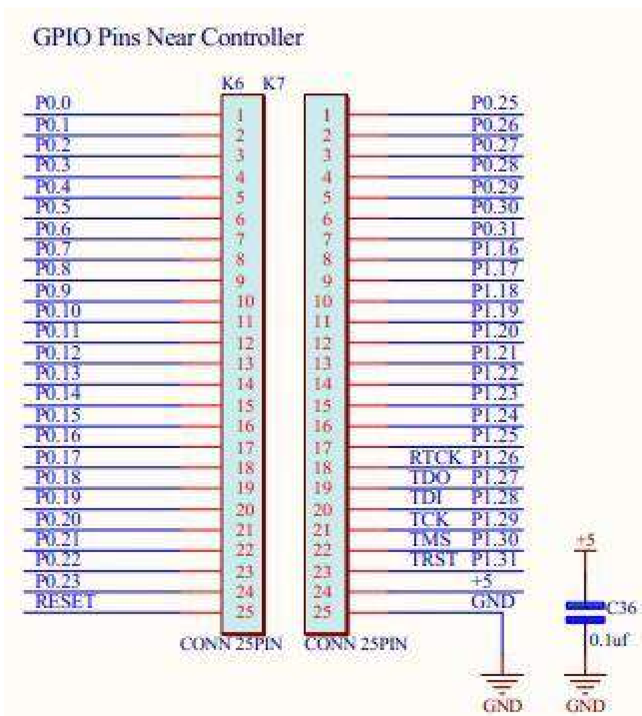
- 2.768 KHz crystal as the RTC clock source
- 12 MHz crystal as the MCU clock source

2.7. MICROCONTROLLER PINOUT



2.8. PORT EXPANDER(ADDITIONAL INPUT/OUTPUT PORTS)

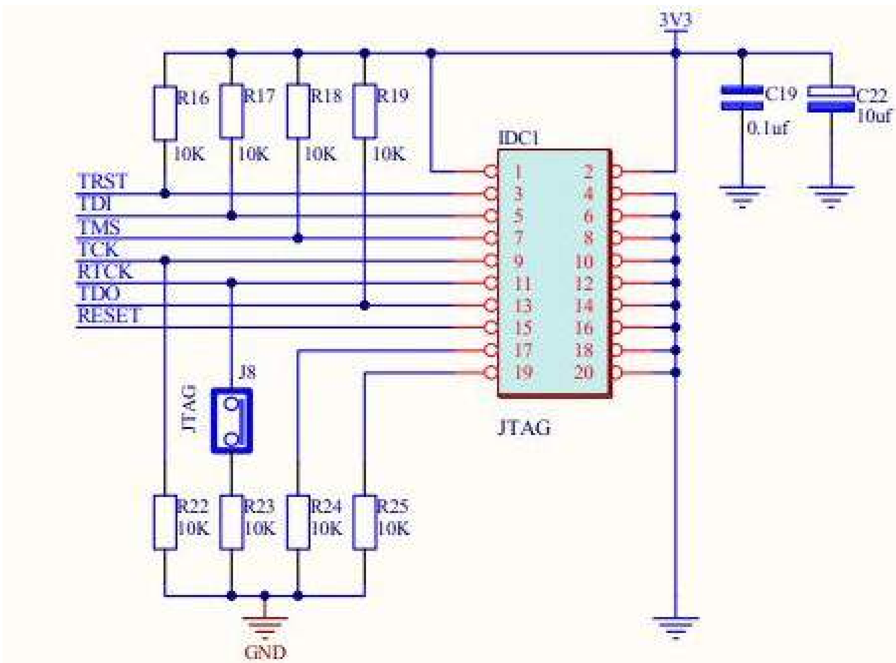




2.9. JTAG CONNECTION

The Joint Test Action Group (JTAG), is an integrated method for testing interconnects on printed circuit boards (PCBs) that are implemented at the integrated circuit (IC) level.

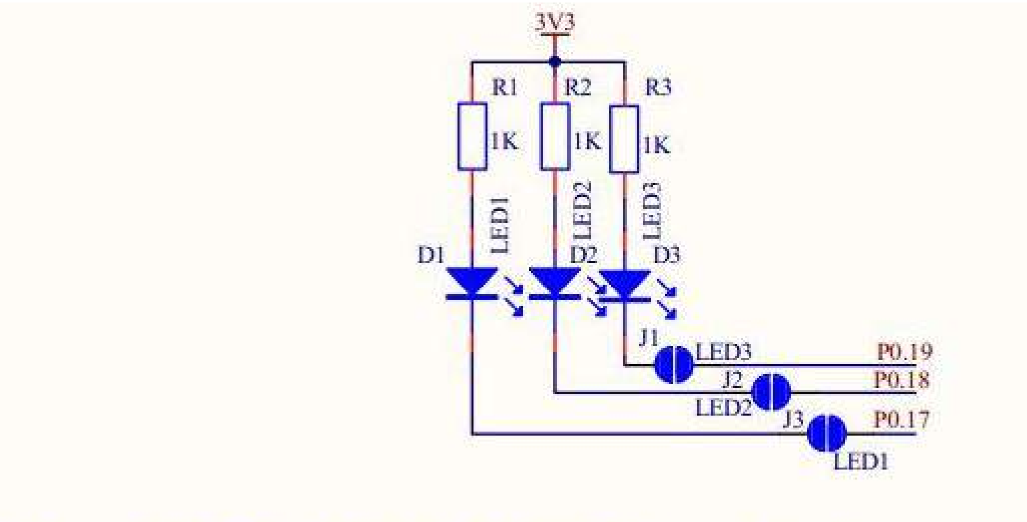
The microcontroller can also be programmed and be used to test the operation of the microcontroller with the JTAG programmer. In order to enable the JTAG programmer to be used, it is necessary to place jumper J8 in the position



2.10. LED INTERFACE

LED's are semiconductor diodes, electronic devices that permit current to flow in only one direction. The diode is formed by bringing two slightly different materials together to form a PN junction. In a PN junction, the P side contains excess positive charge ("holes") while the N side contains excess negative charge ("electrons"). When a forward voltage is applied to the semi conducting element forming the PN junction, electrons move from N area toward P area and holes move from P area toward N area. Near the junction, the electrons and holes combine. As this occurs, energy is released in the form of light that is emitted by the LED. The material used in the semi conducting element of an LED determines its color. LED's are the simplest devices to test port functioning.

LPC 2148 mini development board has 3 SMD LED s connected to port pins P0.17, P0.18 & P0.19 via jumpers J1 , J2 & J3. If any jumper is left open, then the corresponding port pin can be used independently. The LEDs turn ON when the port pins are at logic high state and they get turned OFF when the port pins are at logic low state. Each LED is interfaced via a current limiting resistor.

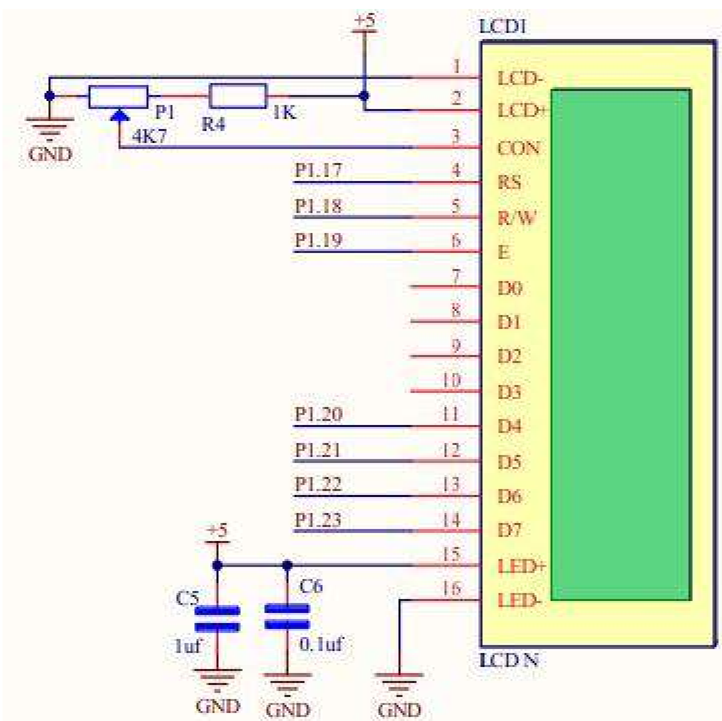


Remove J1,J2 &J3,while P0.19,P0.18 &P0.17 are used for other purpose

Note: *Remove J1,J2 & J3 when P0.19,P0.18 & P0.17 are used for other purpose.*

2.11. LCD INTERFACE

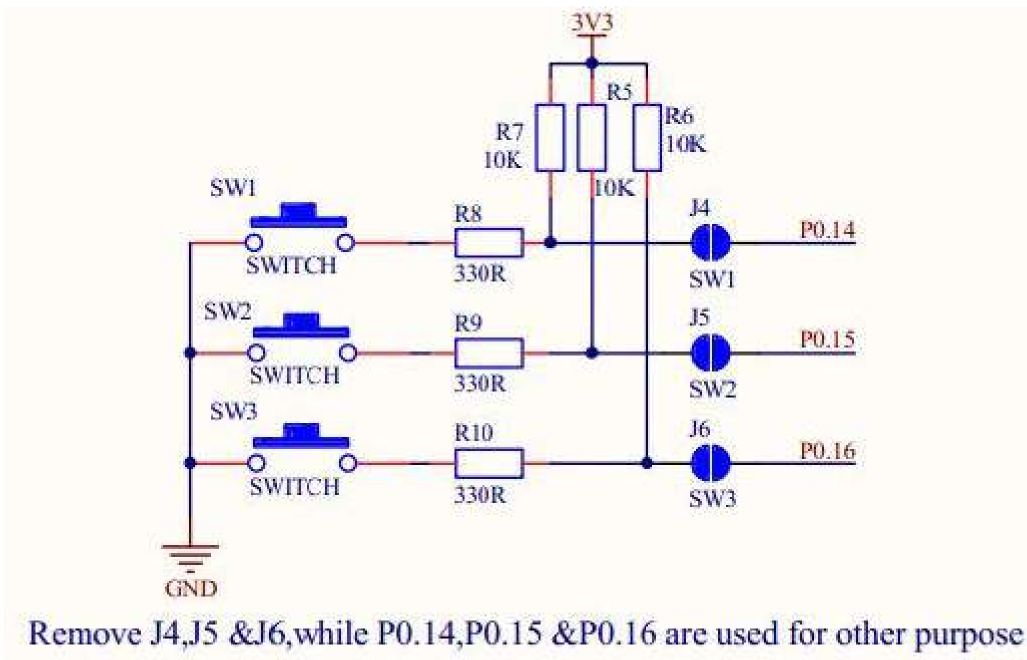
The display is a standard 16x2 LCD which displays 2 lines of 16 characters. Each character is 40 pixels, making it 1280 pixels overall. The display receives ASCII codes for each character at the data inputs (D0–D7). The data is presented to the display inputs by the MCU, and latched in by triggering the E (Enable) input. The RW (Read/Write) line can be tied low (write mode), as the LCD is receiving data only. The RS (Register Select) input allows commands to be sent to the display. RS selects command/data mode. The display itself contains a microcontroller; the standard chip in this type of display is the Hitachi HD44780. It must be initialized according to the data and display options required. The module can be used 4-bit or 8-bit mode. The development board uses 4-bit interface. Data pins are P1.20-P1.23 and control pins are P1.17(RS),P1.18(R/W) and P1.19(E). LCD contrast can be adjusted by using the potentiometer.



2.12. PULL-UP KEY INTERFACING

The simplest input to a microcontroller is a switch or push button. This can operate with just one additional support component, a pull-up resistor. The board has 3 externally pulled up switches (SW1,SW2 & SW3) connected to port pins P0.14, P0.15 & P0.16 via jumpers J4, J5 & J6 respectively. On shorting these jumpers, the switches can be used as general matrix keypad and if left open, then the port pins can be used for other purposes.

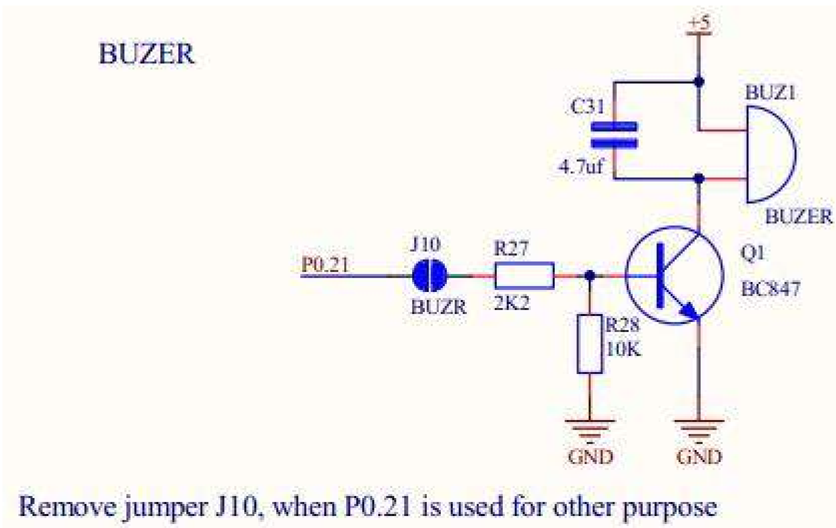
When a switch is open, then the corresponding port pin gets pulled up to 5V by the pull-up resistor &



Note : *Remove J4,J5,J6 when P0.14, P0.15 & P0.16 are used for other purpose.*

2.13. BUZZER INTERFACING

Buzzer is a simple I/O device. Normally we use piezo electric element as buzzer. Buzzer is driven using a simple NPN transistor with biasing. The transistor's base is connected to P0.21 of the microcontroller via jumper J10. If the port pin is configured as output pin and logic high, the transistor will be triggered on which in turn switch on the Buzzer. If logic low is provided, the buzzer will be turned off.



Note : *Remove J10 when P0.21 is used for other purpose*

2.14. UNIVERSAL ASYNCHRONOUS RECEIVER TRANSMITER

LPC2148 has two UART modules namely UART0 and UART1. It has only asynchronous(no clock connection) mode of operation. For UART0, transmission & reception pins are respectively P0.0 & P0.1. UART1 communicates through P0.8(TXD1) and P0.9(RXD1).

In the mini development board, UART0 can communicate through

1. Serial port via MAX232
2. USB port via CP2102
3. RMC connector (GND,TXD0,RXD0, Vout) in 3.3V/5V levels
4. Zigbee connectors on either side of the board

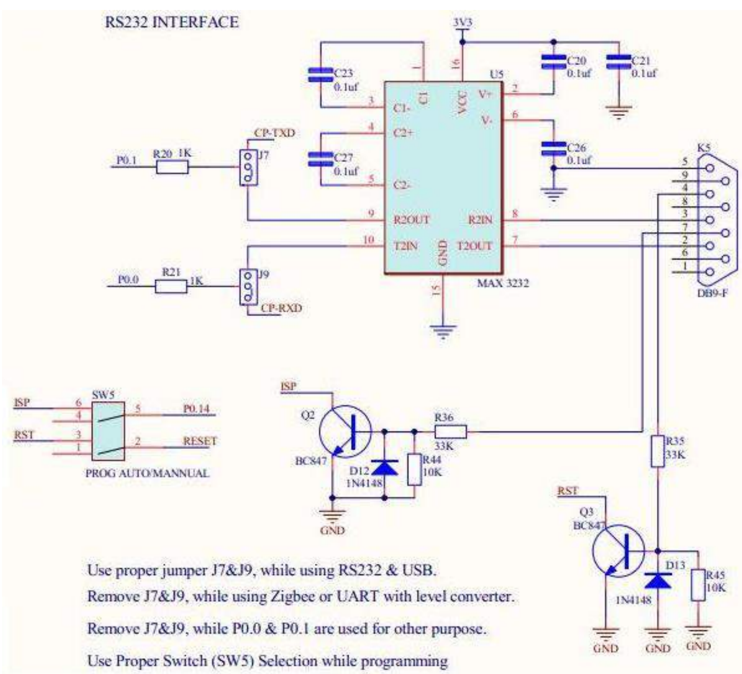
UART0

UART0 communication via USB/Serial port is selected using jumpers J7 and J9. Through RMC connector, UART0 can be used in two voltage levels of 3V3/5V which can be selected by jumper J16.

Note: *While using UART0 for communication, PROGRAMMER switch should be in manual mode*

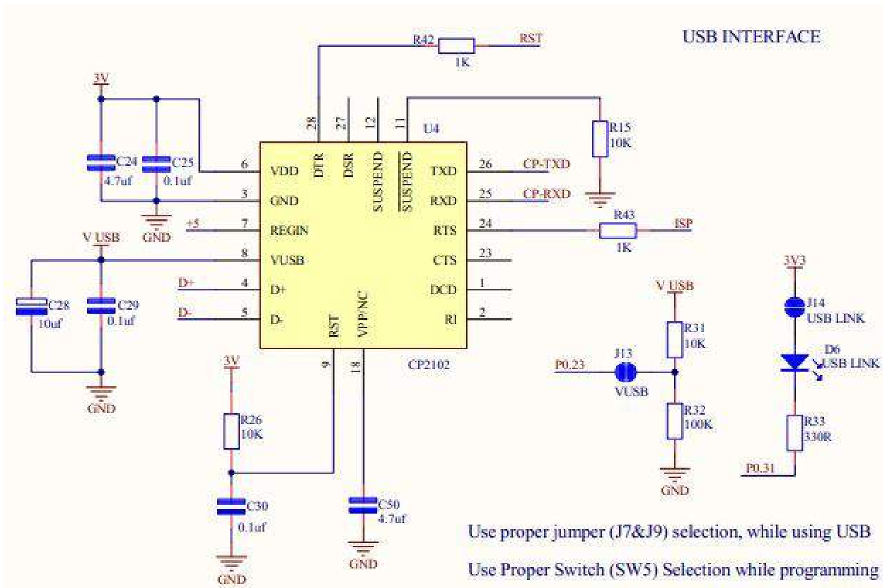
- **RS232 Interface via DB9 connector**

The RS232 interfacing is done by using the serial driver IC MAX 232 and a DB9 connector. The MAX232 is an IC that converts signal from RS232 serial port to signal suitable for use in TTL compatible digital logic circuit. The MAX 232 is a dual driver/ receiver and typically converts RX, TX, CTS and RTS



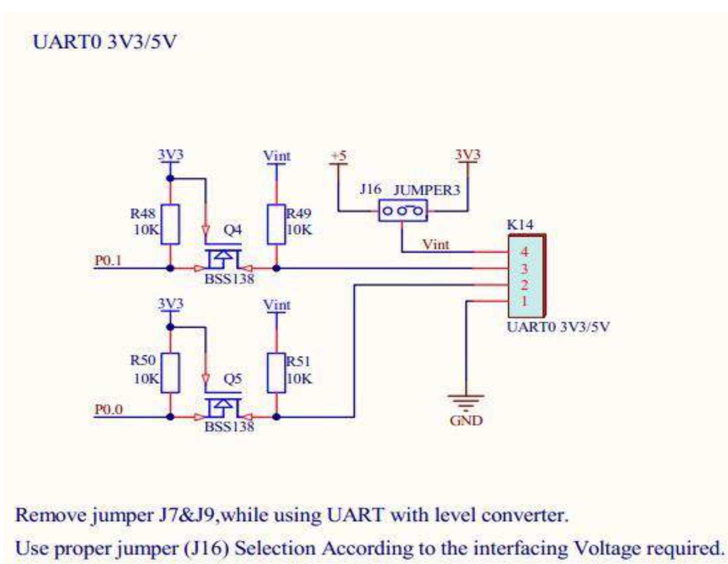
- **USB Interface using CP2102**

CP2102 is a highly-integrated USB-to-UART Bridge Controller providing a simple solution for updating RS-232 designs to USB using a minimum of components and PCB space. CP2102 includes a USB 2.0 full-speed function controller, USB transceiver, oscillator, EEPROM, and asynchronous serial data bus (UART) with full modem control signals in a compact 5 x 5 mm MLP-28 package. No other external USB components are required.

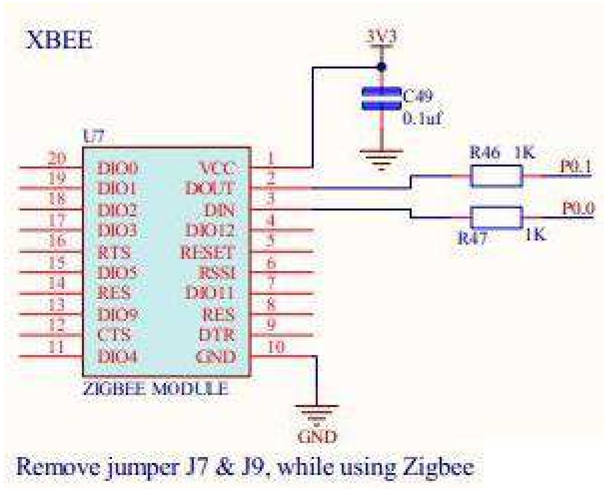


- **3V3 /5V UART0 via RMC connector**

Through RMC connector, UART0 can communicate in two voltage levels of 3.3 & 5 V. This makes it possible to interface both 3.3 & 5 V TTL modules. The voltage level is selected using jumper J16.



- **Zigbee Module Interface**

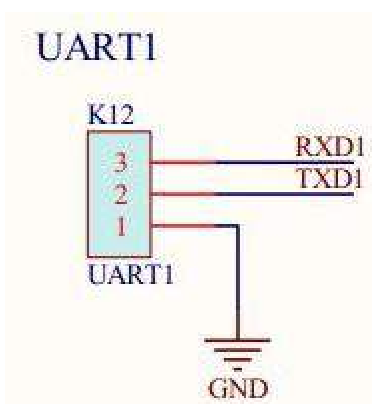


Note: *Remove jumpers J7&J9, while using Zigbee*

UART1

UART1 of LPC 2148 communicates through connector K12 on the board that has 3 pins namely *GND,TXD1 & RXD1*.

- TXD1 - Transmission pin of UART1 (P0.8)
- RXD1 - Reception pin of UART1 (P0.9)
- GND - Common ground



USING REALTERM IN PC

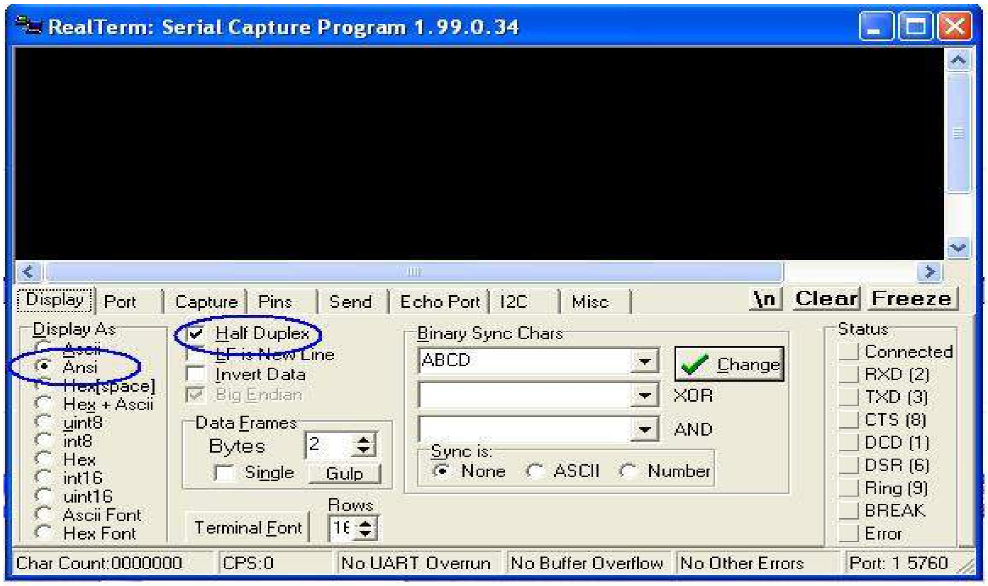
Real term is a testing, analyzing and simulation tool for serial communication protocols. It allows us to monitor communication between two serial devices or to test the serial communication of a single devices. **Realterm** can be download by (download)

Steps for creating RealTerm in PC

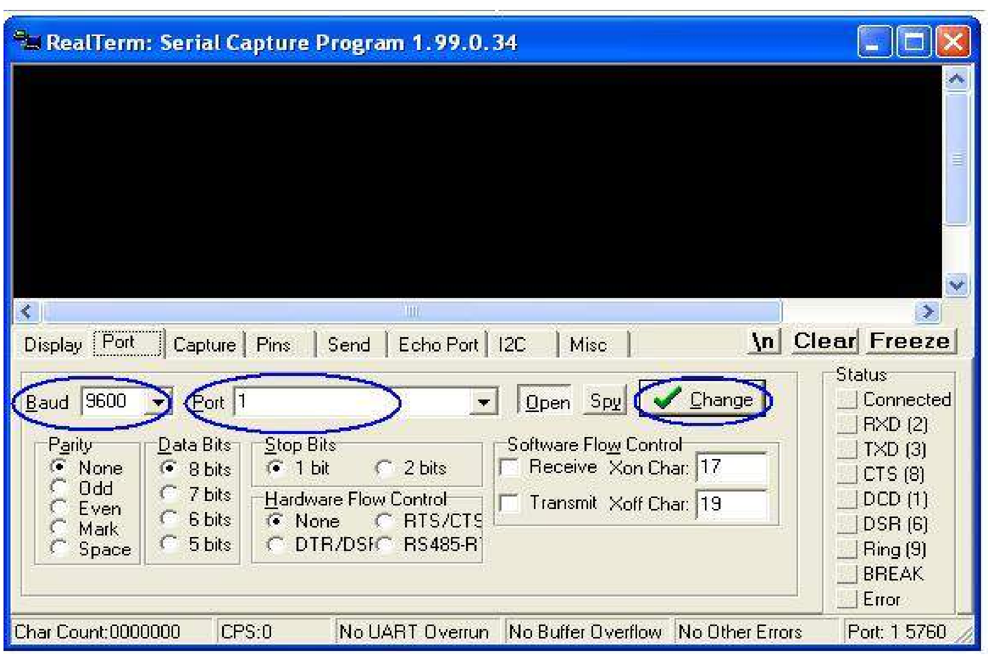
The serial data transmitted through USART can be viewed on a PC using a Windows tool for Serial Port Communication called Realterm.

Step 1: All program ⇒ RealTerm ⇒ realterm

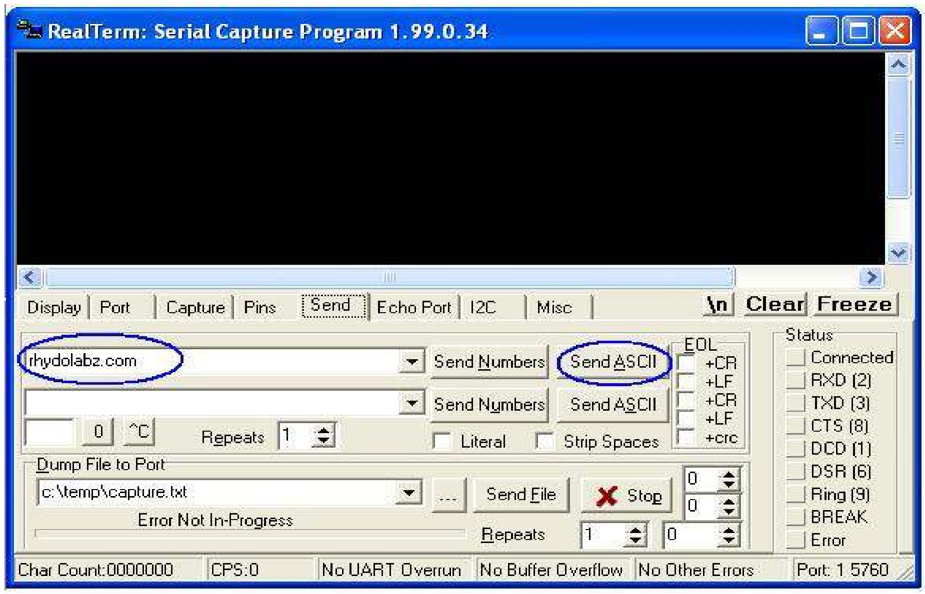
Step 2: ***Display Tab***- Here the output text format selected is ANSI and Half Duplex mode is enabled to view the data sent by the user.



Step 3: ***Port Tab***-To test the connection - make sure the **Open** button is pressed, Select required baud rate and the “Port” dropdown here, select the number of your COM port and then press the **Change** button.

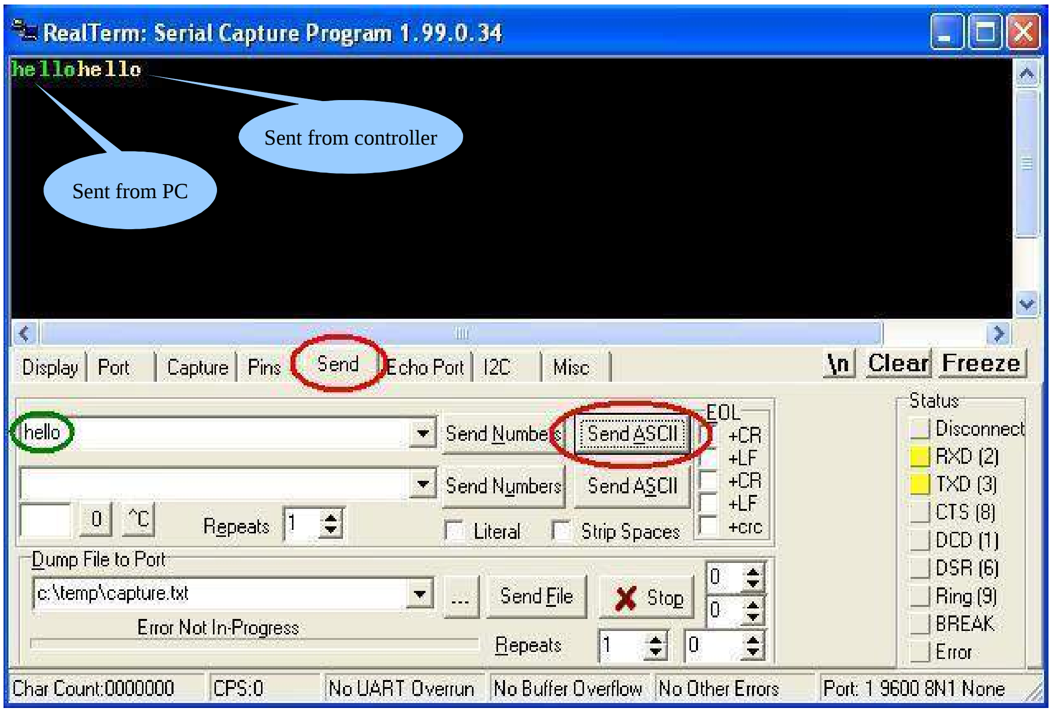


Step 4: Send Tab- Insert the desired data to be transmitted and press "Send ASCII" button.



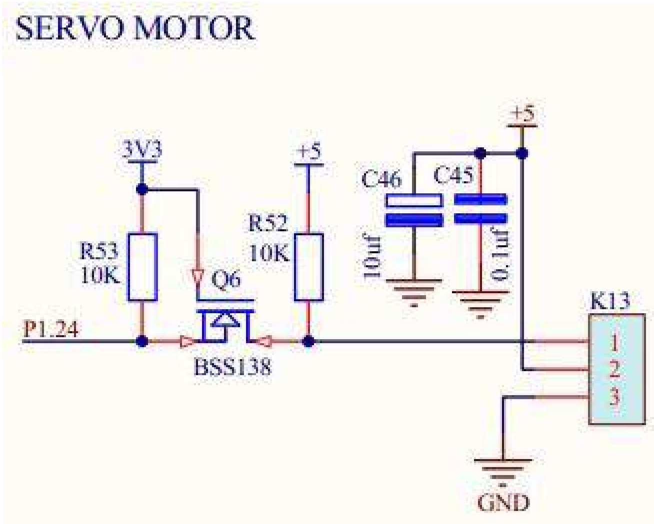
Step 5 : The output after data transmission to the controller is shown in the following diagram. The text sent by user and controller is highlighted by callouts in the figure.

Step 6: To check reception, go to Send option, type the string in the space provided(encircled in green) and click **Send ASCII** button. The first "hello" in green colour is transmitted from PC & that in yellow colour is retransmitted by the controller



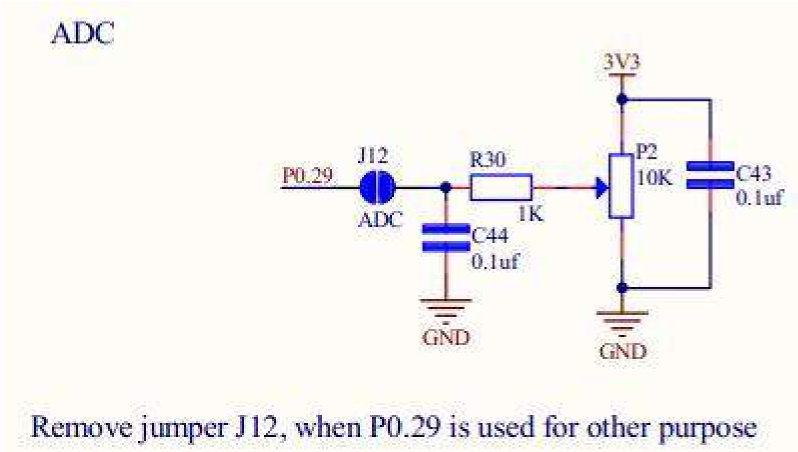
2.15. SERVO MOTOR INTERFACING

Servos are small mechanical motorized devices whose sole purpose is to rotate a tiny shaft attached to a servo wheel in a specified position. Servos are controlled by sending a pulse width signal from an external electronic device that generate PWM signal values. PWM signal send to the servo are translated into position values by electronics inside the servo. In the mini development board, servo is connected to P1.24.



2.16. ANALOG-TO-DIGITAL CONVERTER

LPC 2148 mini development board has a potentiometer connected to its ADC pin P0.29 (channel2) via jumper J12.

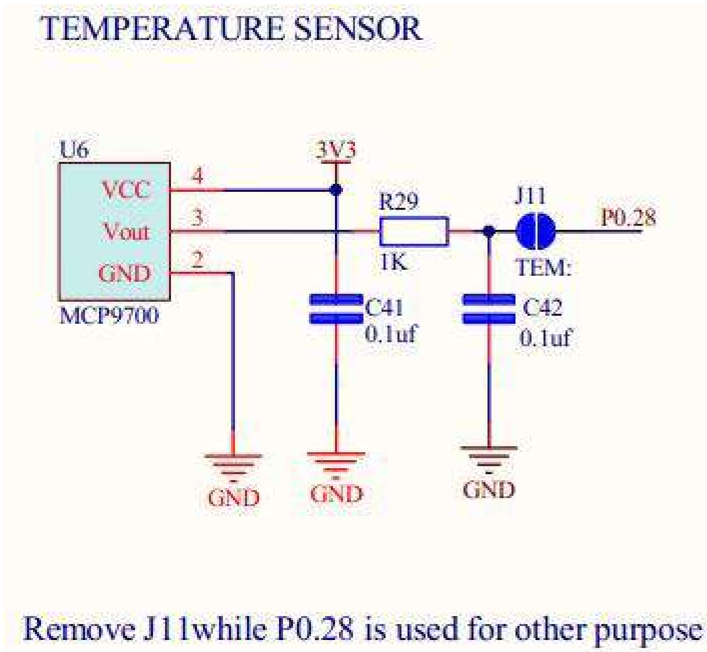


.

Note : *Remove J12 when P0.29 is used for other purpose*

2.17. TEMPARATURE SENSOR INTERFACE

MCP 9700 temperature sensor can be used to measure temperature. It is connected to P0.28(channel 1) via jumper J11.



Note: *Remove J11 when P0.28 is used for other purpose*

SOFTWARE DEVELOPMENT

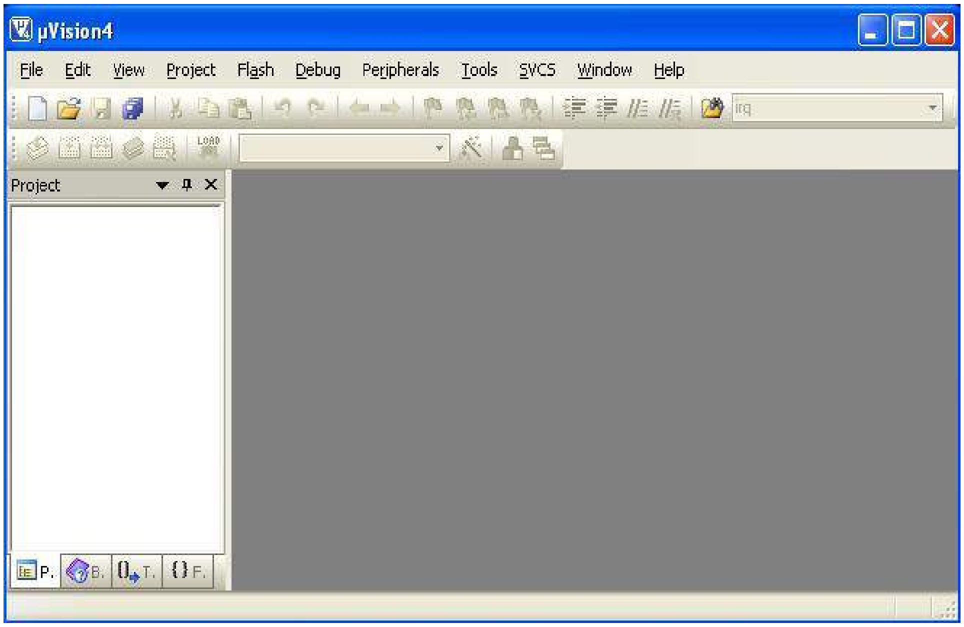
3.1. TOOLS AND SOFTWARE

FAMILIARIZATION OF Keil uVision4

1. Open **Keil uVision4** from start menu or Desktop shortcut
2. Create new Project File and Select CPU.
3. Create New Source Files.
4. Add Source Files to the Project.
5. Set Tool Options for Target Hardware.
6. Create a HEX File.
7. Build Project and Generate Application Program Code.

Launch Keil uVision4

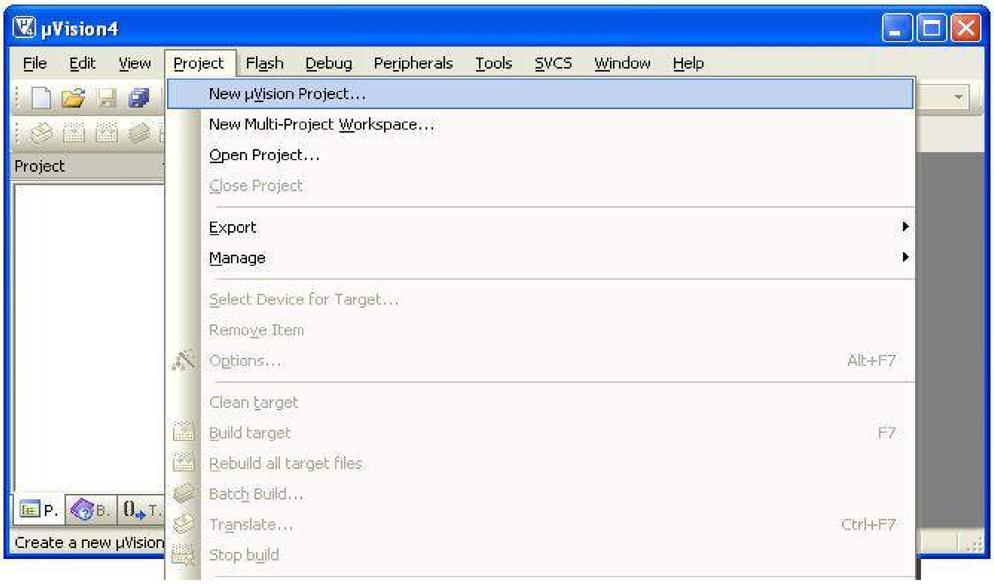
The Keil uVision4 window opens as shown below



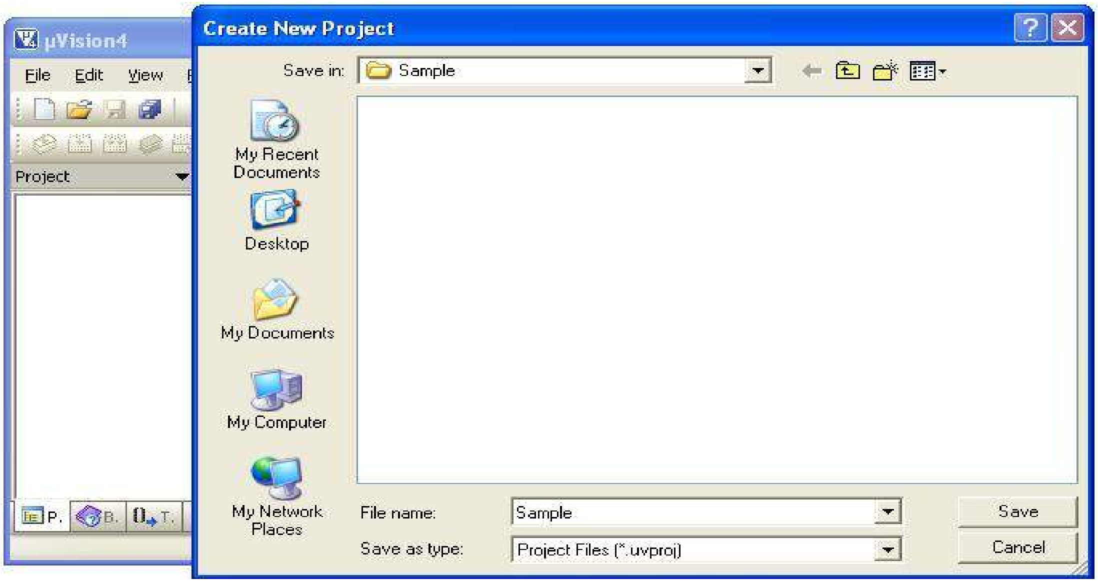
Create new Project File and Select CPU.

This section provides a step-by-step tutorial that shows you how to create a simple Keil uVision4 project.

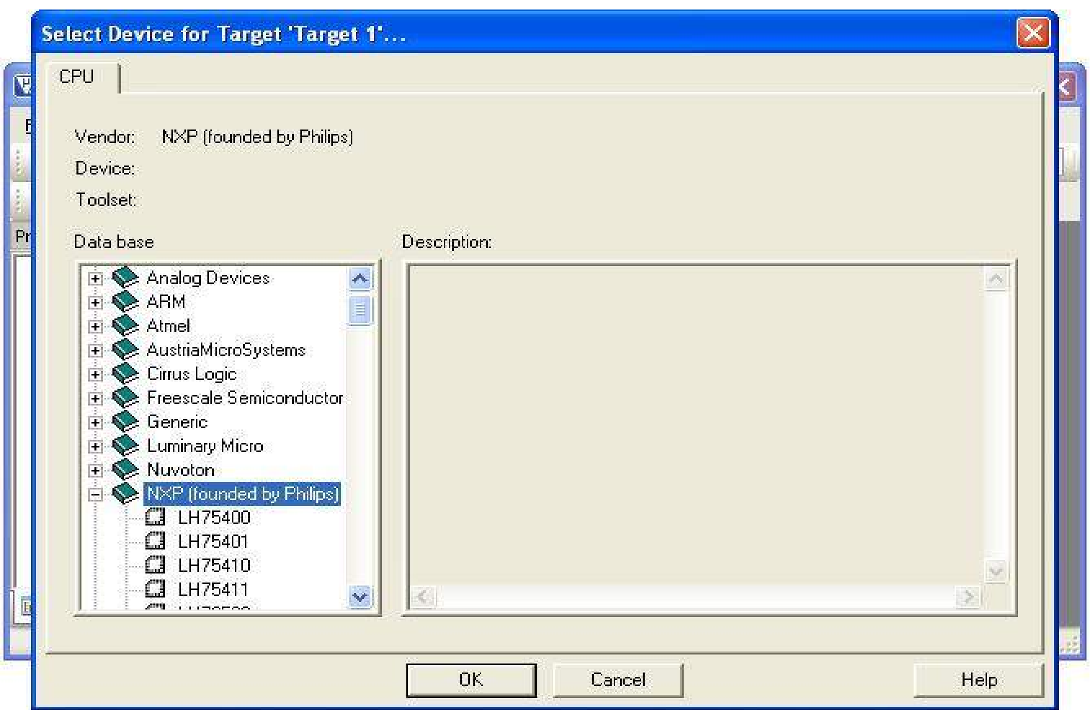
Step 1: To create a new project, select *Project > New uVision Project* from menu bar



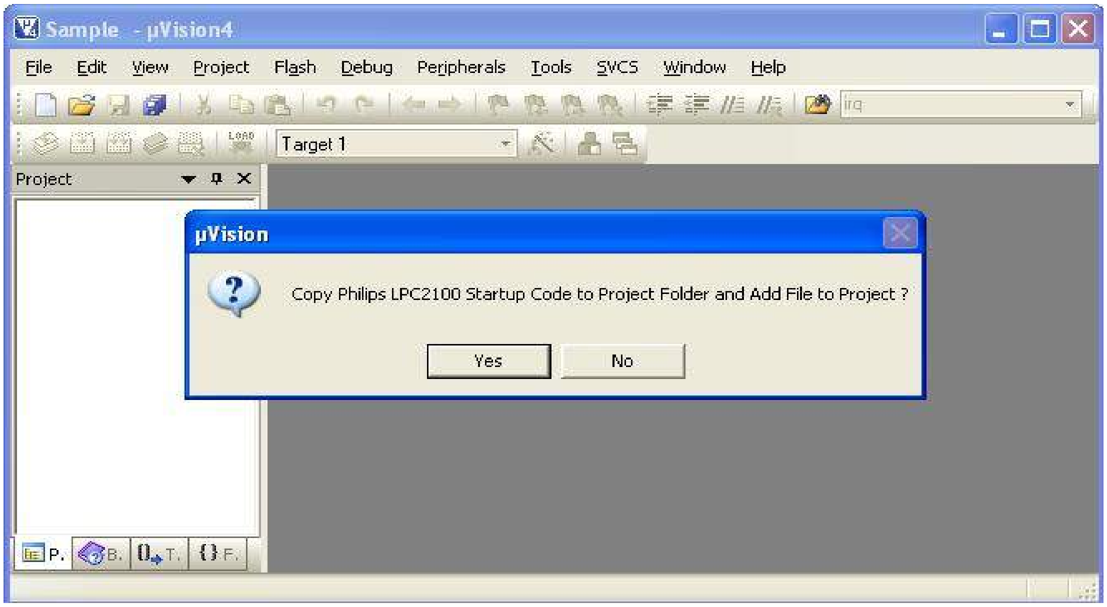
Step 2: Save the project in a suitable location with appropriate name



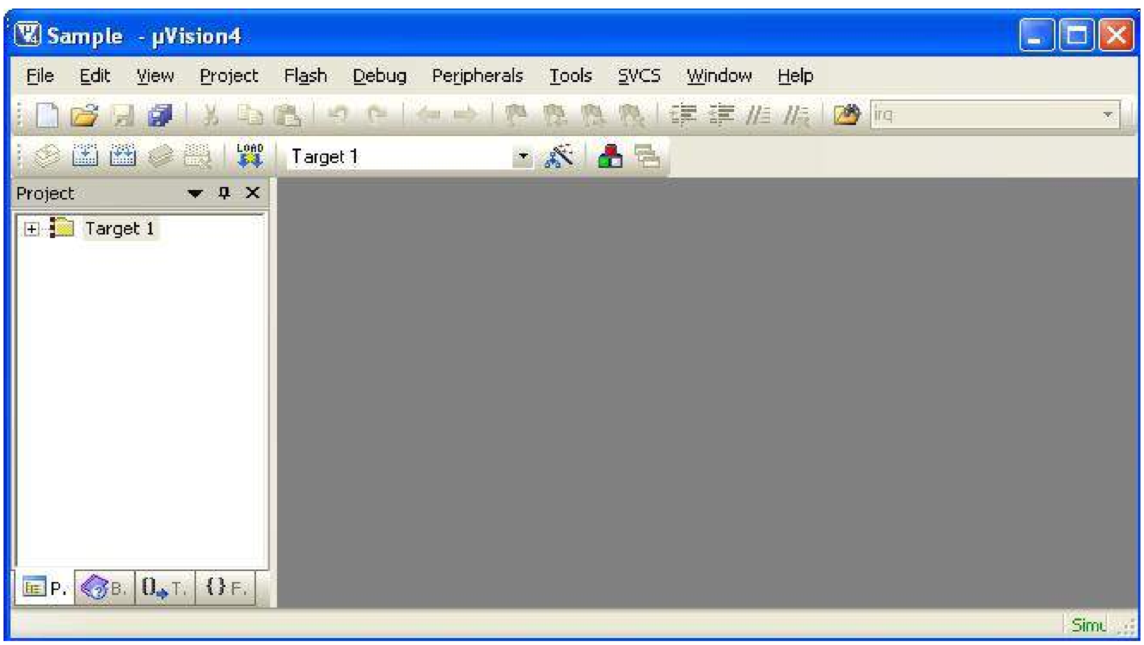
Step 3: The following window opens. Select LPC 2148 (listed under NXP) from the drop-down list



Step 4: Click 'Yes' for the following question to copy the Startup code to Project folder and add file to project

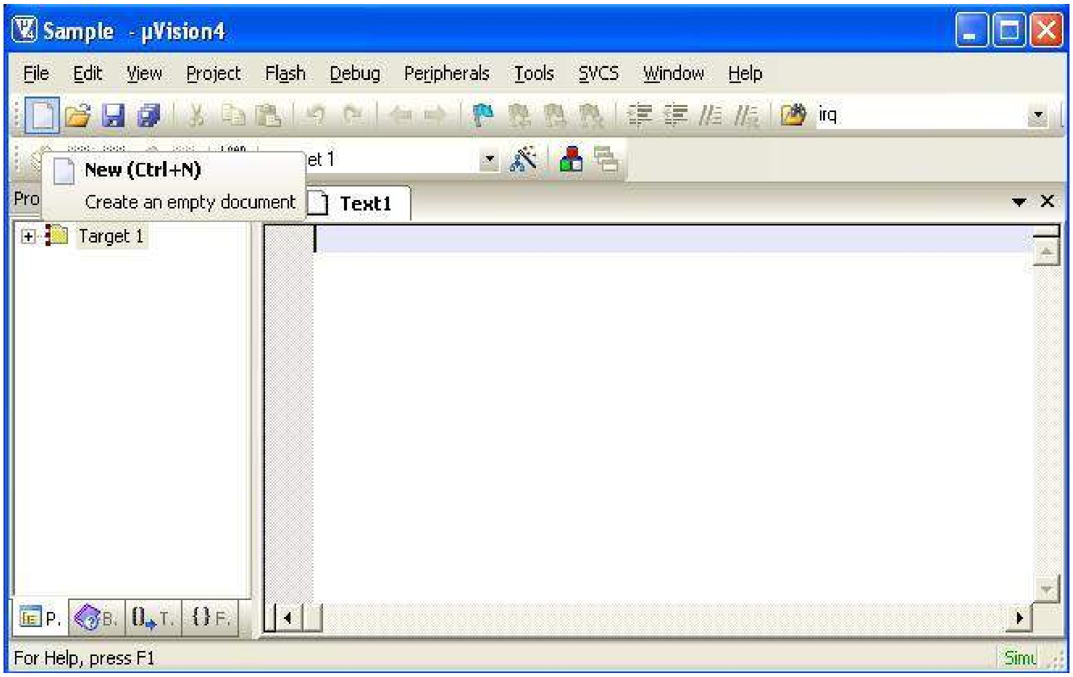


Step 5: This creates a target to the project



Create New Source Files

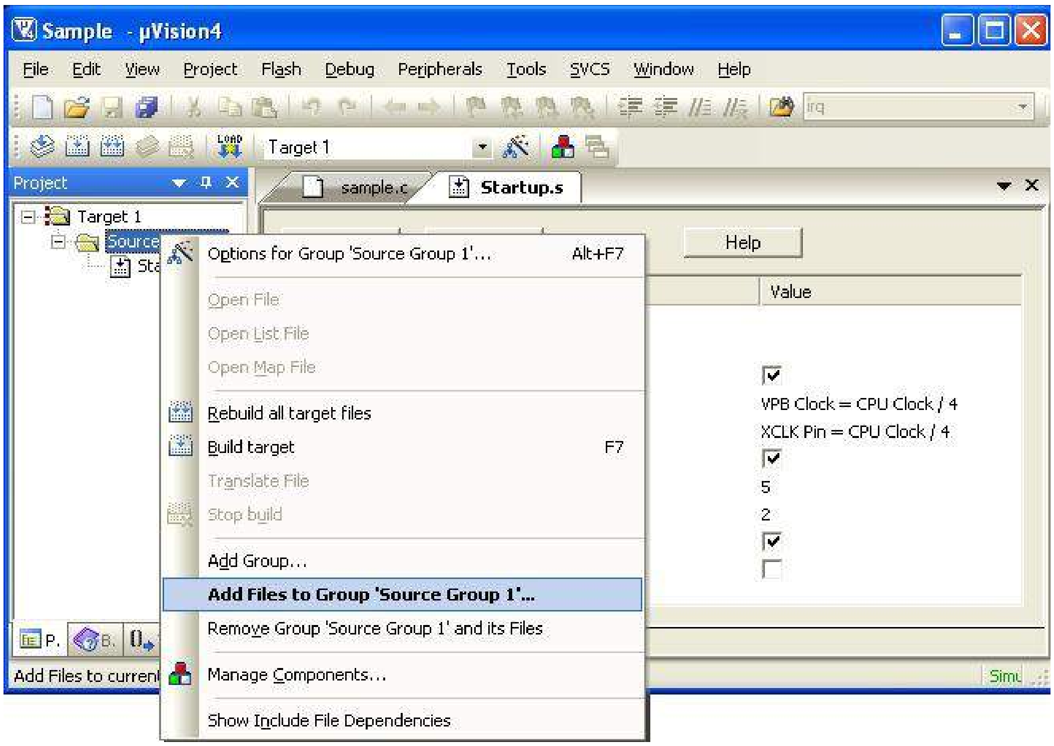
Step 6: Create a new file either by clicking the ***New File icon***, or by selecting ***File > New*** or using keyboard shortcut ***CTRL + N***



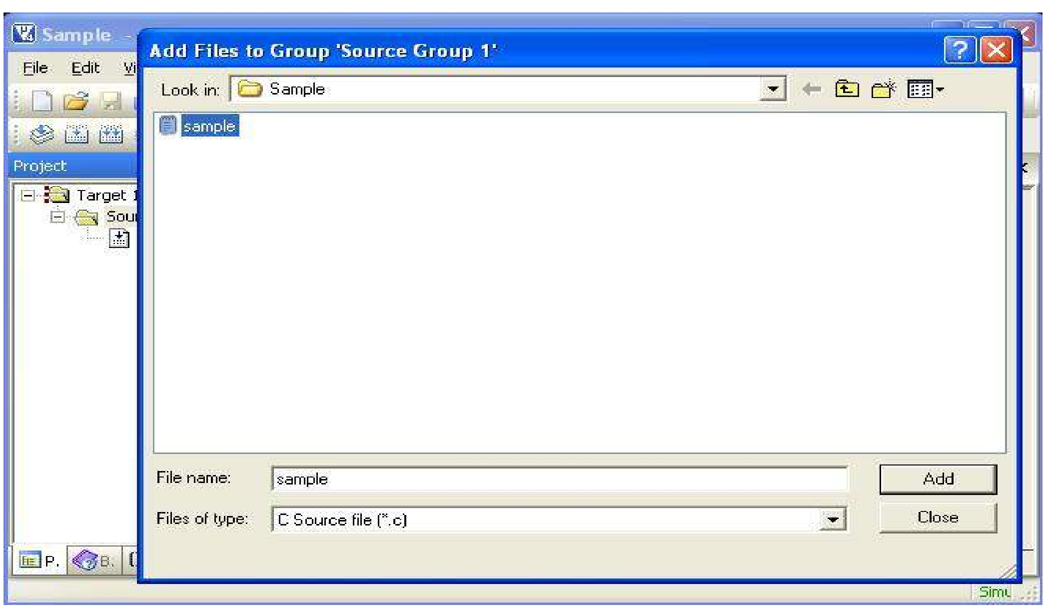
Step 7: Save the file with ***.c extension*** in the project folder

Add Source Files to Project

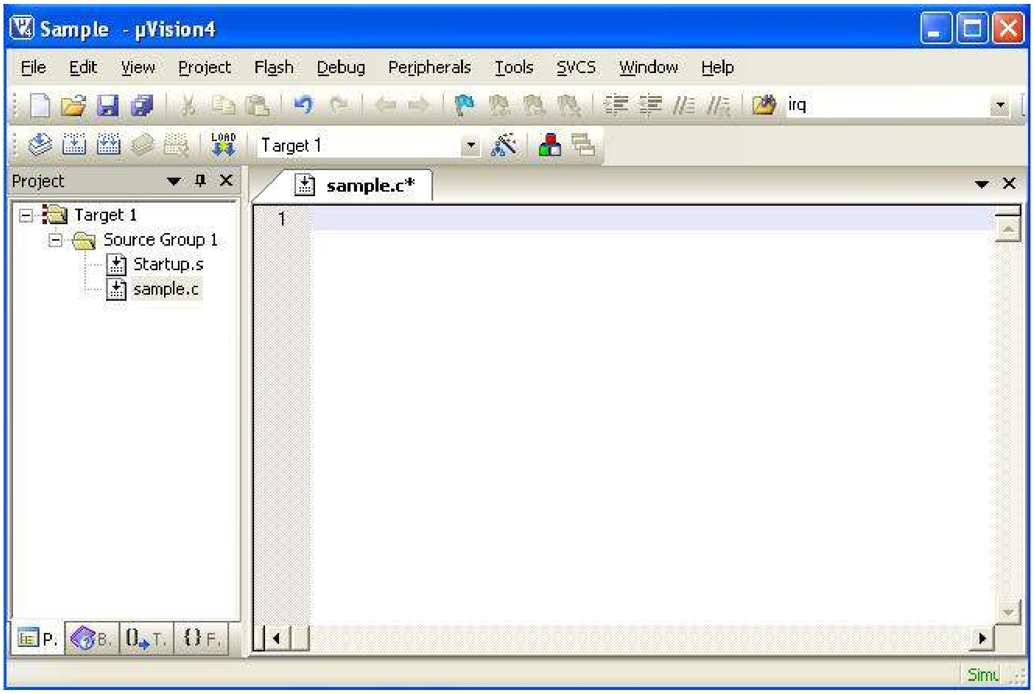
Step 8: Right click Source Group 1 to add C file to source group



Step 9: Select the C file created and click Add

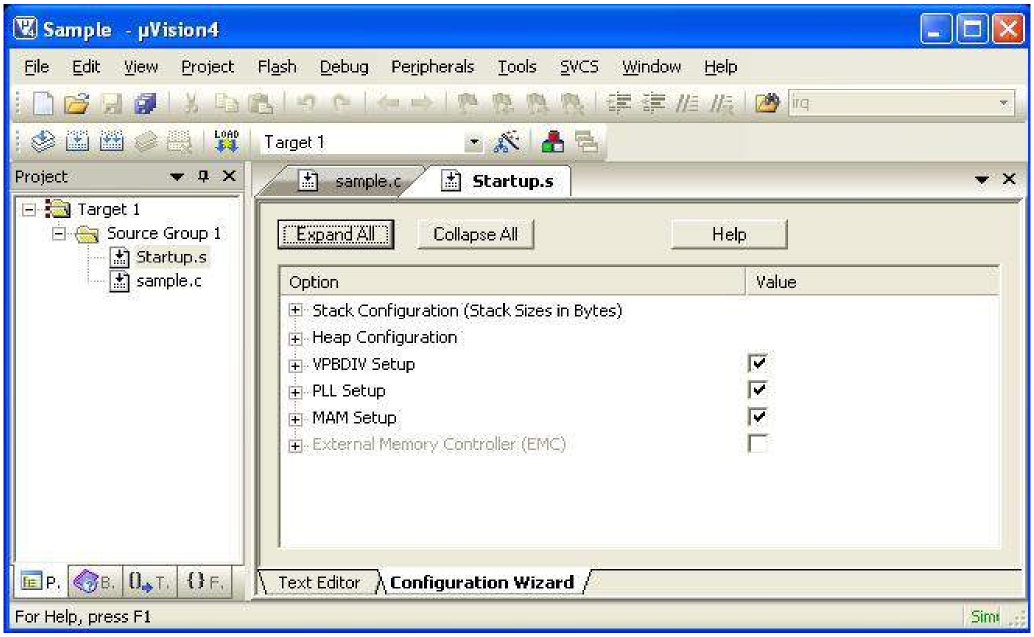


Step 10: Now the c file gets added to the Source

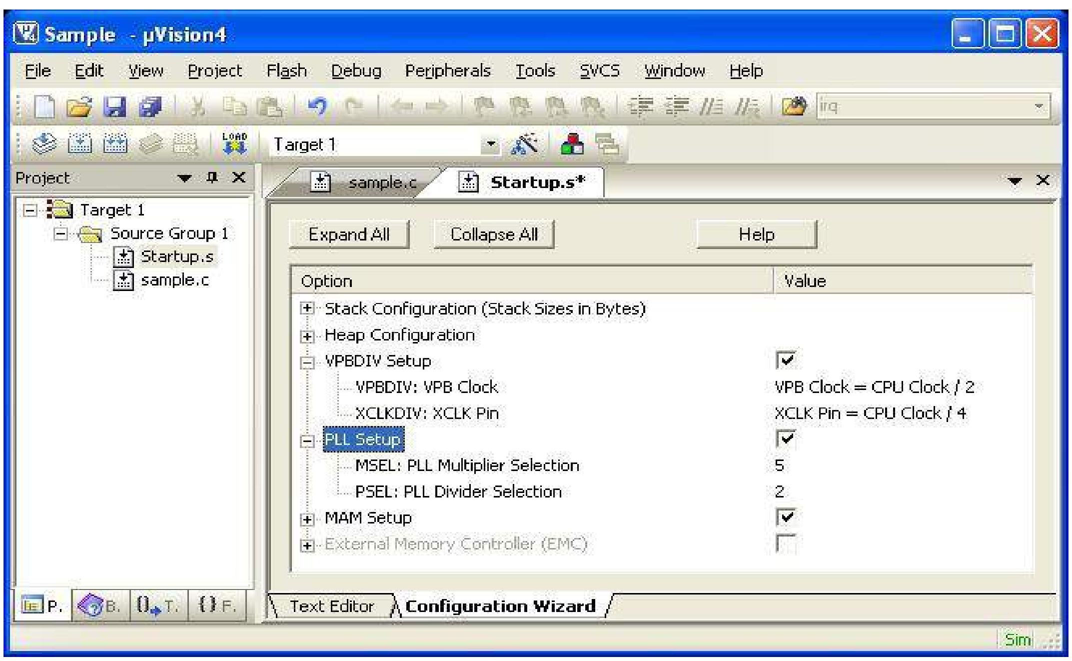



```
#include<lpc21xx.h>
void Delay(unsigned long val);

int main()
{
IO0DIR  = 0XFFFFFFFF;        /* Set Port0 as output                          */
while(1)                     /* Infinite loop                                */
{
IO0SET = 0X000E0000;       /* Set P0.17,P0.18,P0.19 at logic high state    */
Delay(1000000);            /* Delay of 100ms                               */
IO0CLR = 0X000E0000;       /* Set P0.17,P0.18,P0.19 at logic low state     */
Delay(1000000);
}
}

void Delay(unsigned long val)
{
while(val>0)
{
val--;
}
}
```

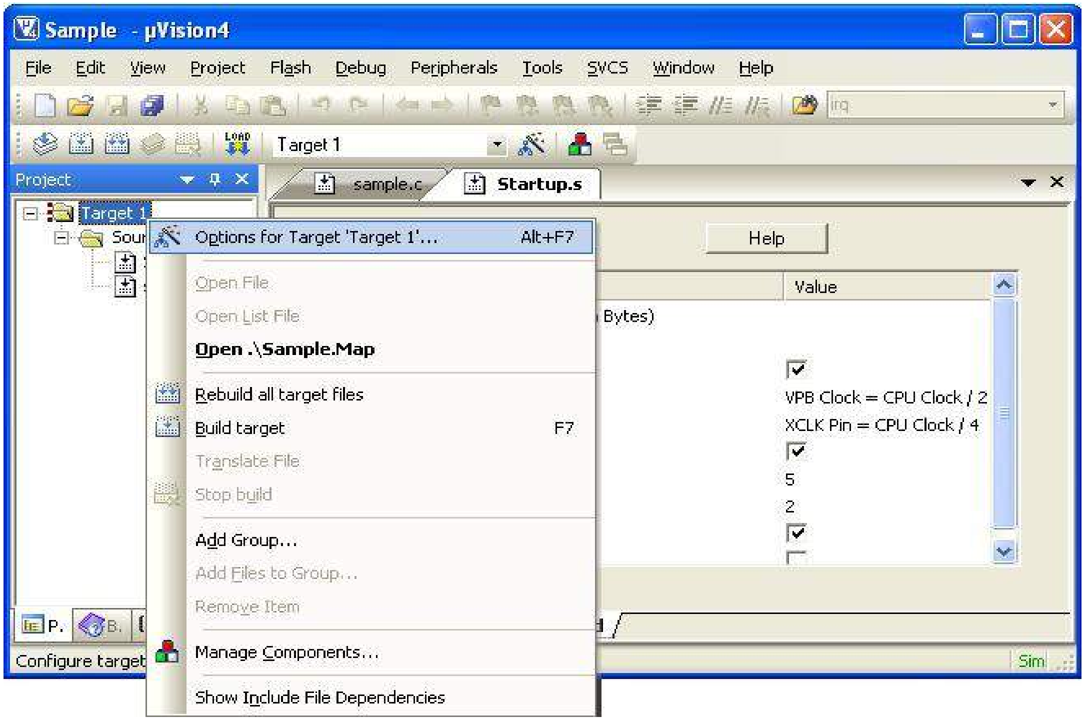
Step 11: Double click on ***"Startup.s"*** to open the configuration window



Step 12: Set the options as shown below and save. The ***PLL setup*** is done for ***12MHz*** crystal. The divider &multiplier must be selected such that the PLL output is **30 MHz ((12/2)*5).**If crystal frequency schanged, thenthese values must be Changed accordingly

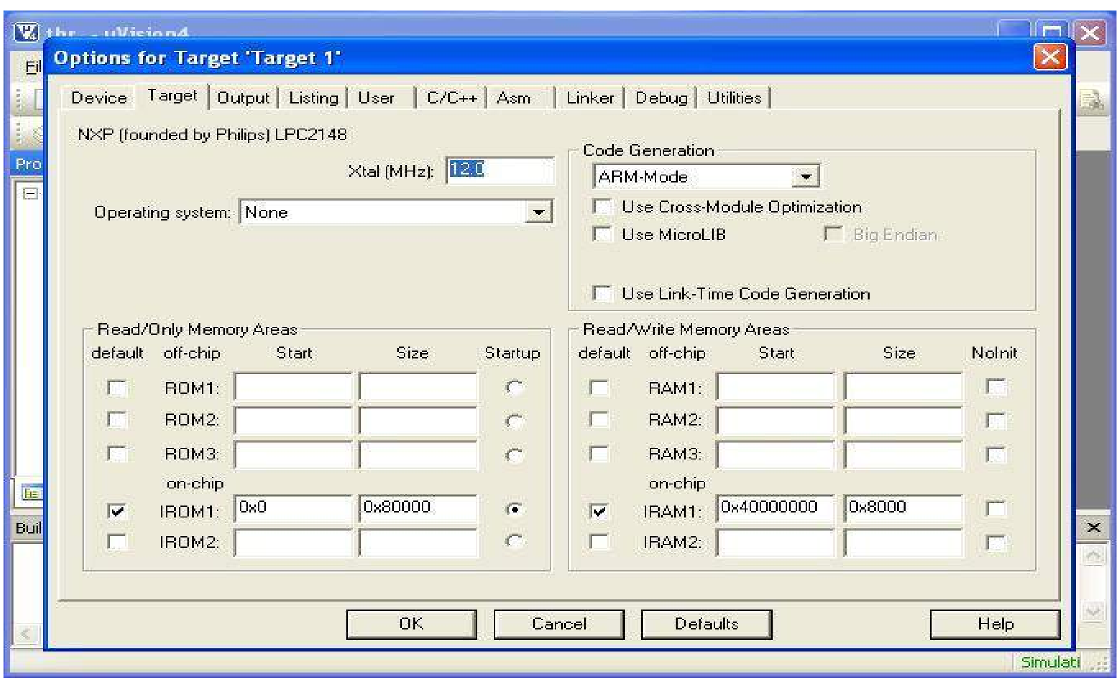


Step 13: Right click on Target1 to set target file options. You can also do this by using the icon on ***'Build toolbar'*** or ***Project > Options for Target 'Target 1***

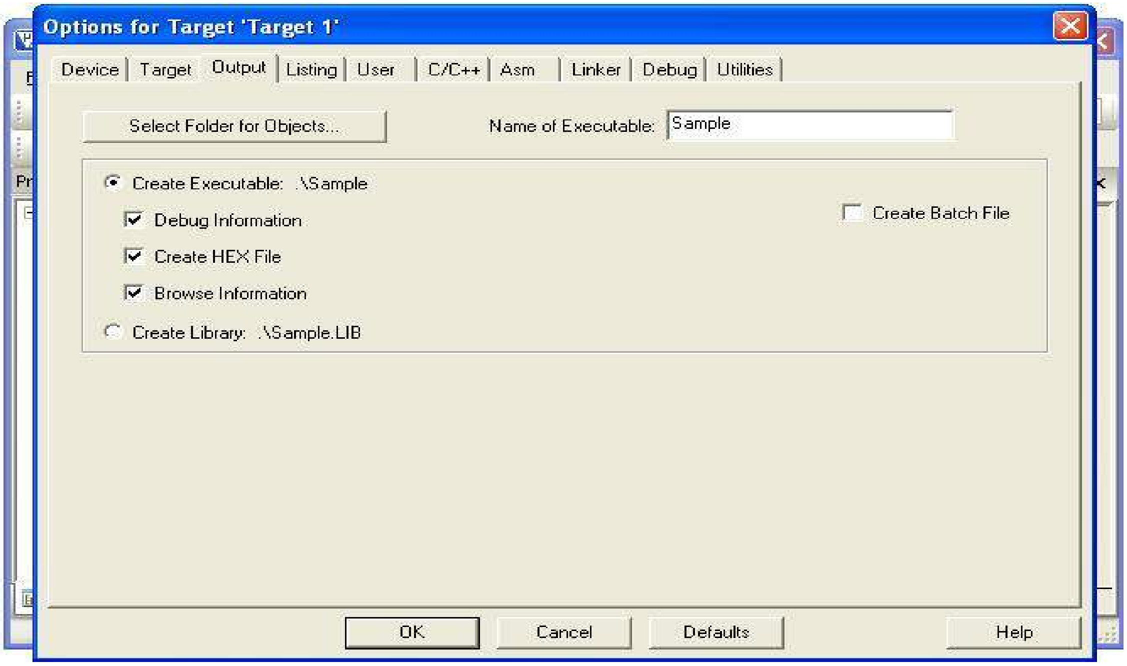


Step 14: Configure Target, Output and Linker options as shown below

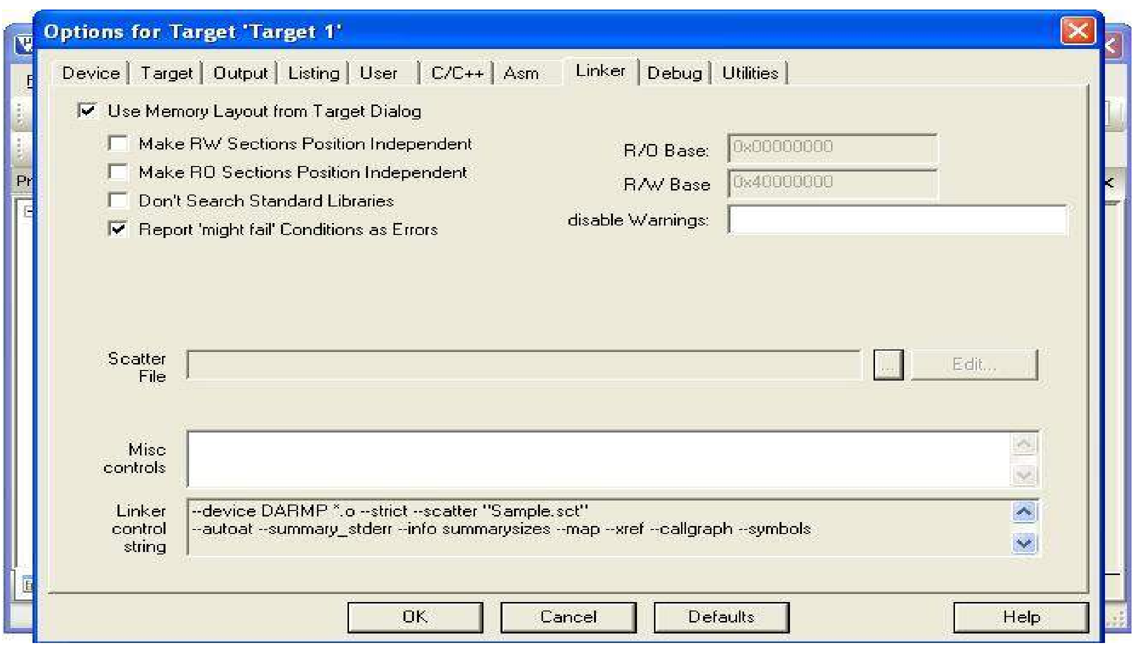
<u>Target</u>



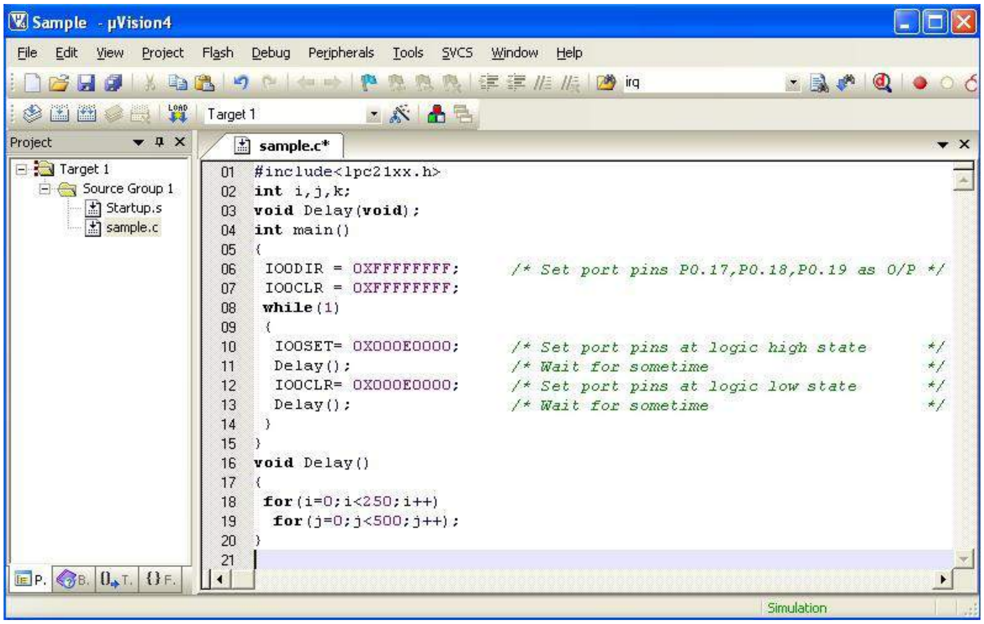
Output



Linker



Step 15: Type the code

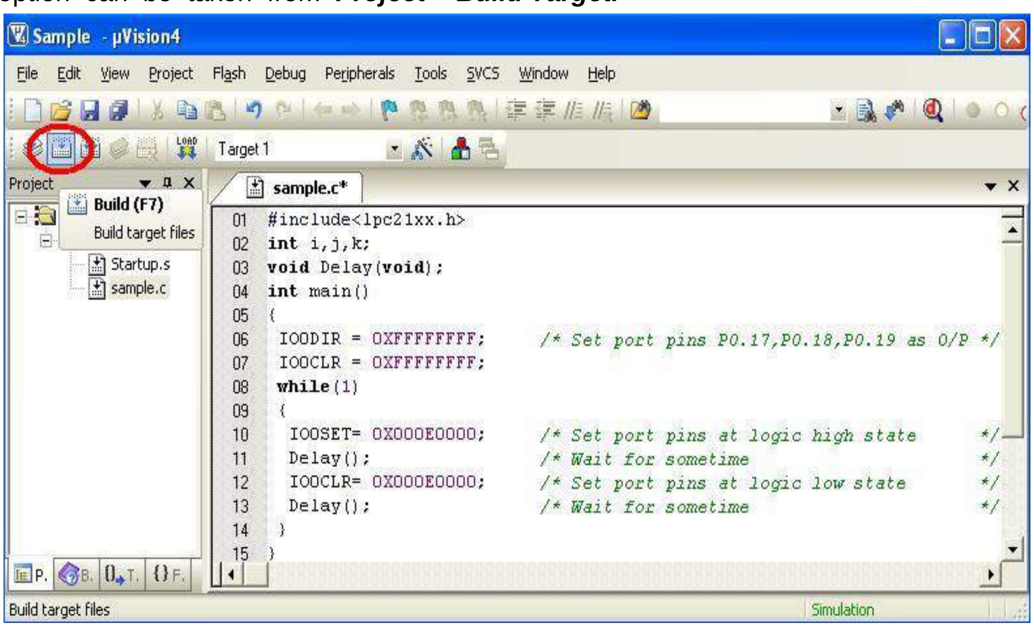


Create HEX File

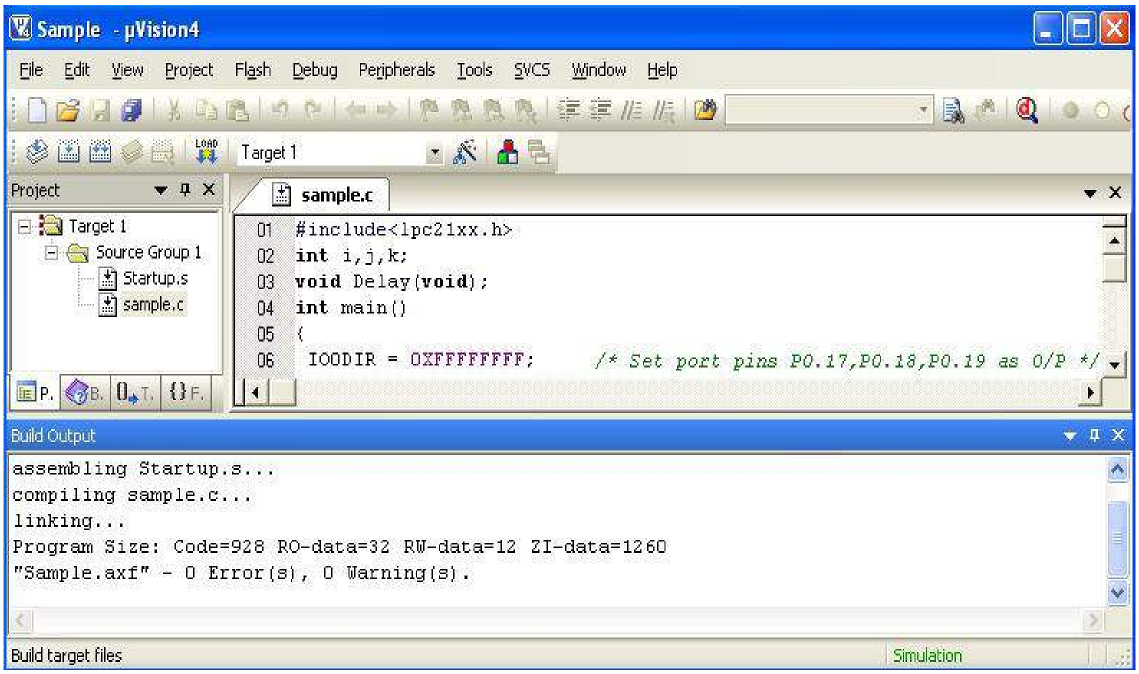
Step 16: Click the ***build icon*** (encircled in figure) to build the project. Errors (if any) get listed in the Build output window. Correct them and build again. On successful building, the hex file will be generated in the project folder

Build Project

Build option can be taken from ***Project > Build Target.***



Now the hex file of the program will be generated in the working folder in the name of the project we created. When you build an application, Keil uVision4 will display errors and warning messages in the **Build page/Build output**. Double clicking a message line opens the corresponding error in the correct location in the Keil uVision4 editor window. In this example the Hex file is **Sample.hex.** This file will be available in the folder Sample.



SETTING UP ARM LPC2148 mini

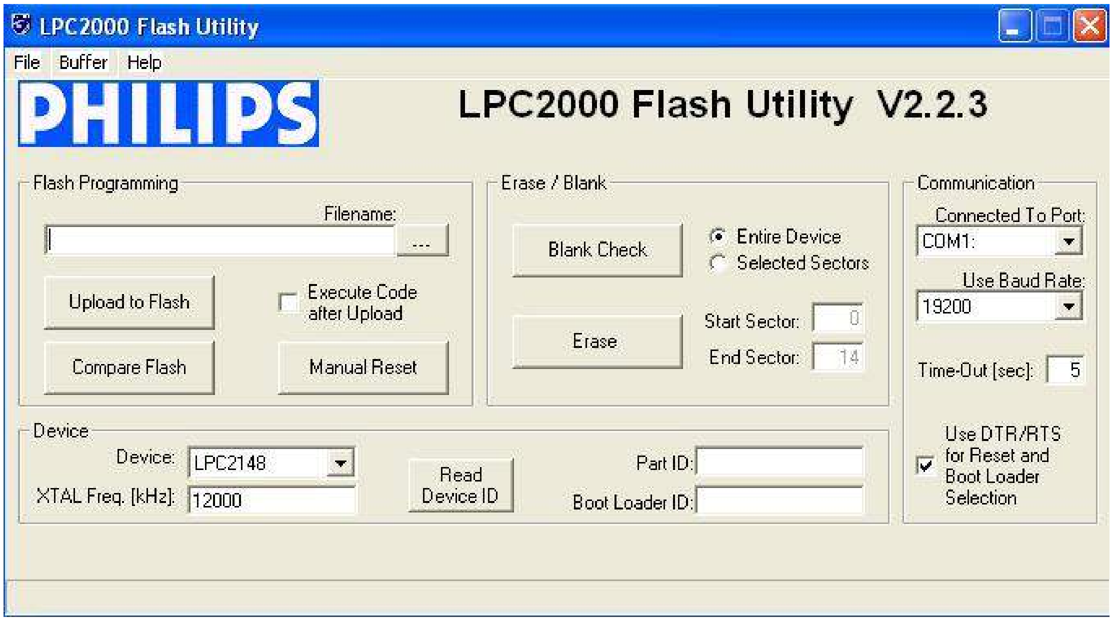
Now the code can be flashed to the controller

In the board, do the following jumper connections

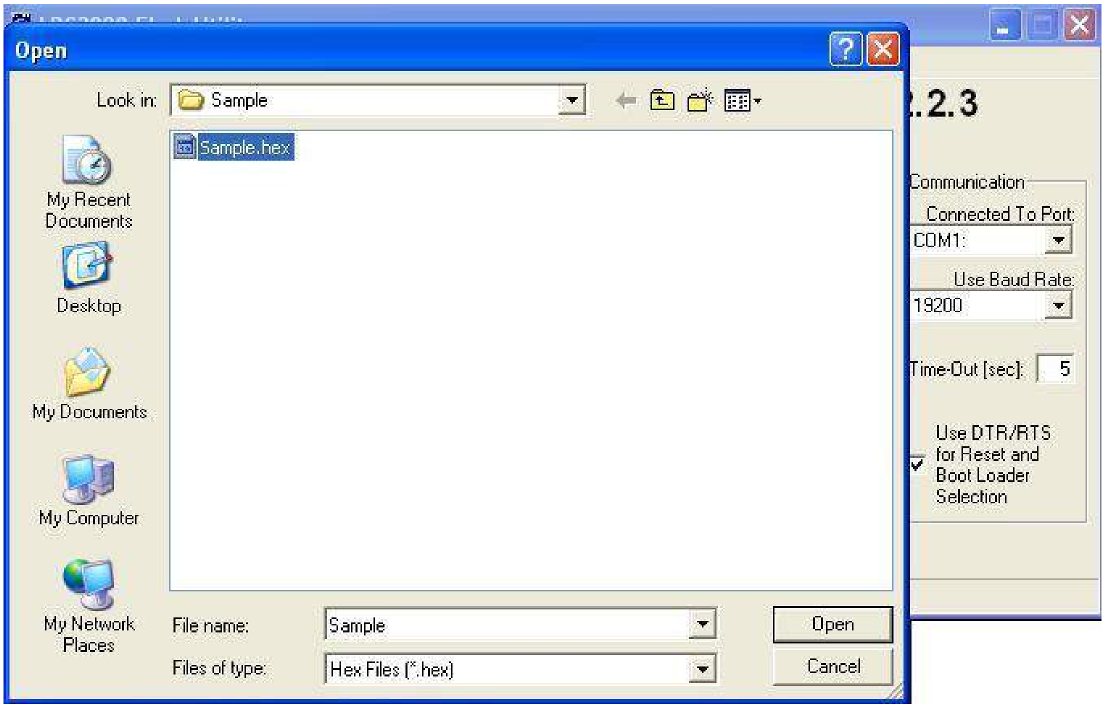
- Select the power source as USB cable or DC source
- Select USB or serial port using jumpers J7 & J9 for flashing the code
  Both jumpers on bottom side means USB programming
  Both jumpers on top side means RS232 serial programming
- Select mode using programmer switch
  In manual mode, insert ISP jumper and press reset button before programming
  In auto mode, leave it open and proceed to programming

PROGRAMMING STEPS

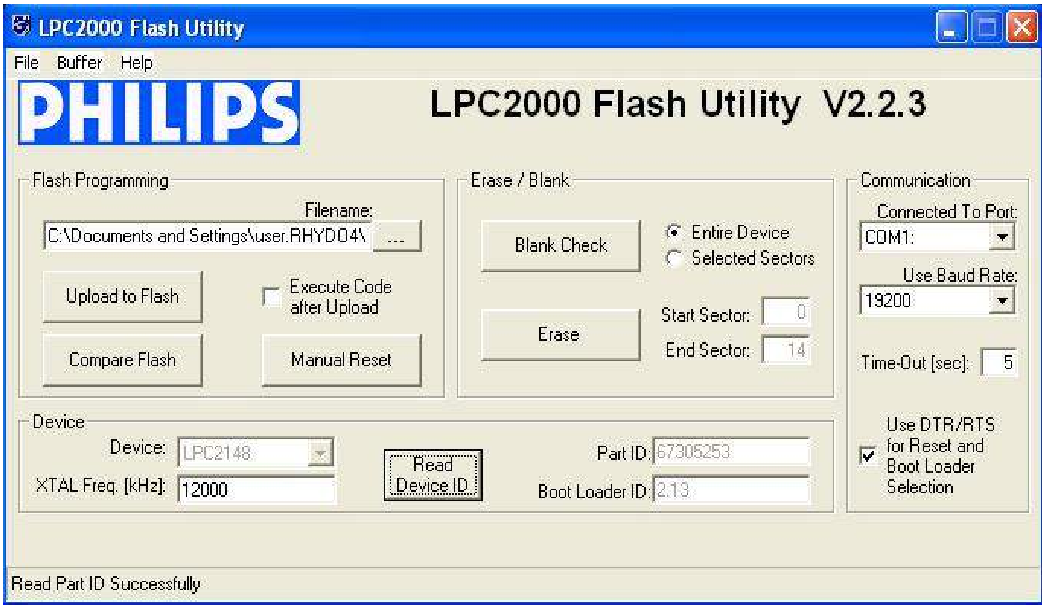
Now power up the board. The power LED(red LED on the board) glows. Open Flash Utility. Select the correct COM port and suitable baud rate



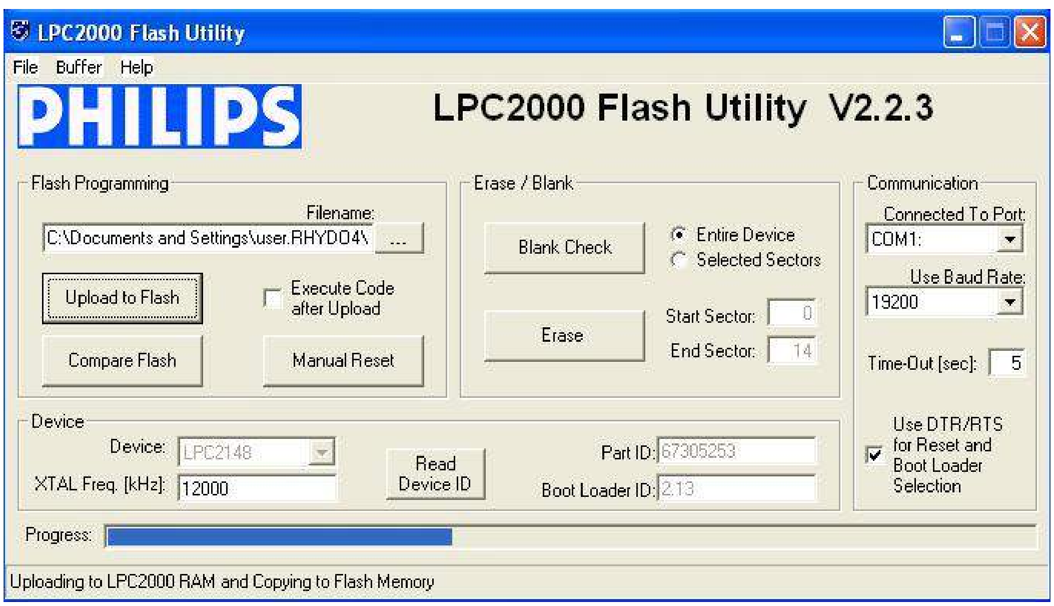
Step 17: Open the desired hex file



Step18: Read device ID If auto mode is selected, simply clicking the '**Read device ID'** button will read the ID. But in manual mode, press reset switch and click



Step 19: Click 'Upload to Flash'



I/O DISTRIBUTION

4.1. THE PIN DISTRIBUTION OF ARM LPC2148 DEVELOPMENT BOARD MINI

PIN No	NAME	TYPE	THE I/O ASSIGN OF LPC 2148 Mini Development Board
1	P0.21	I/O	BUZZER/PWM5
2	P0.22	I/O	UNUSED
3	RTXC1	-	CRYSTAL
4	P1.19	I/O	LCD(E)/TRACEPKT3
5	RTXC2	-	CRYSTAL
6	VSS	-	GROUND
7	VDDA	-	3V3
8	P1.18	I/O	LCD(R/W),TRACEPKT2
9	P0.25	I/O	AD0.4
10	P0.26	I/O	D+
11	P0.27	I/O	D-
12	P1.17	I/O	LCD(RS)/TRACEPKT1
13	P0.28	I/O	TEMPERATURE SENSOR/AD0.1
14	P0.29	I/O	POTENTIOMETER/AD0.2
15	P0.30	I/O	AD0.3/EINT3
16	P1.16	I/O	TRACEPKT0
17	P0.31	I/O	UP_LED
18	VSS	-	GROUND
19	P0.0	I/O	TXD0/PWM1/ZIGBEE/ MAX 232(T2IN)/CP2102(RX)
20	P1.31	I/O	TRST
21	P0.1	I/O	RXD0/PWM3/ZIGBEE/ MAX232(R2OUT)/CP2102(TX)
22	P0.2	I/O	SCL0
23	VDD	-	3V
24	P1.26	I/O	RTCK
25	VSS	-	GROUND
26	P0.3	I/O	SDA0
27	P0.4	I/O	SCK0/AD0.6
28	P1.25	I/O	EXTIN0
29	P0.5	I/O	MISO0/AD0.7

30	P0.6	I/O	MOSI0
31	P0.7	I/O	PWM2/SSEL0
32	P1.24	I/O	SERVO MOTOR/TRACECLK
33	P0.8	I/O	TXD1/PWM4
34	P0.9	I/O	RXD1/PWM6/EINT3
35	P0.10	I/O	RTS1
36	P1.23	I/O	LCD(D7)
37	P0.11	I/O	SCL1
38	P0.12	I/O	
39	P0.13	I/O	
40	P1.22	I/O	LCD(D6)
41	P0.14	I/O	SDA1/EINT1/SWITCH
42	VSS	-	GROUND
43	VDD	-	3V3
44	P1.21	I/O	LCD(D5)
45	P0.15	I/O	EINT2/SWITCH SW2
46	P0.16	I/O	EINT0/SWITCH SW3
47	P0.17	I/O	SCK1/LED1
48	P1.20	I/O	LCD(D4)/TRACESYNC
49	VBAT	-	3V3
50	VSS	-	GROUND
51	VDD	-	3V3
52	P1.30	I/O	TMS
53	P0.18	I/O	MISO1/LED2
54	P0.19	I/O	MOSI1/LED3
55	P0.20	I/O	EINT3/SSEL1
56	P1.29	I/O	TCK
57	RESET	-	RESET
58	P0.23	I/O	Vbus
59	VSSA	-	GROUND
60	P1.28	I/O	TD1
61	XTAL2	-	CRYSTAL
62	XTAL1	-	CRYSTAL
63	VREF	-	3V3
64	P1.27	I/O	TD0

TECHNICAL SUPPORT

If you are experiencing a problem that is not described in this manual, please contact us. Our phone lines are open from 9:00 AM – 5.00 PM (Indian Standard Time) Monday through Saturday excluding holidays. Email can be sent to support@rhydolabz.com

DISCLAIMER

Contact Us

Rhydo Technologies (P) Ltd.
(An ISO 9001:2008 Certified R&D Company)
Golden Plaza, Chittoor Road, Cochin – 682018, Kerala, India.
Tel: 0091 484 2370 444, 2371 666, Cell: 0091- 99466 70444
Fax: 0091 484 237 0579, Email: info@rhydolabz.com, sales@rhydolabz.com
For more information please visit: www.rhydolabz.com