Advanced Electronics Systems, Bengaluru

ALS-PJTBRD-ARMCTXM3-01

PROJECT BOARD OF ARM CORTEX M3 – LPC1768

User Manual

ADVANCED ELECTRONIC SYSTEMS #143, 9th MAIN ROAD, LAGGERE CROSS, NEAR RAJAGOPAL NAGAR OLD POLICE STATION, PEENYA INDUSTRIAL AREA, BANGALORE-560058
PHONE: 91-80-41625285/41539484. E-mail: sales@alsindia.net URL: www.alsindia.net

USER MANUAL

CONTENTS

1. INTRODUCTION

ALS-PJTBRD-ARMCTXM3-01 is a project board of ARM CORTEX M3 based micro controller LPC1768. Board has on board interfaces like 16×2 LCD, external interrupt circuit, Internal ADC circuit, USB Host 2.0, 2 UART's, reset circuit, ISP circuit, JTAG port, Trace port, CAN interface, general purpose output LED's and 4 numbers of 20 pin single row berg sticks for external interface through port lines.

The ARM Cortex-M3 is a general purpose 32-bit microprocessor, which offers high performance and very low power consumption. The Cortex-M3 offers many new features, including a Thumb-2 instruction set, low interrupt latency, hardware divide, interruptible/continuable multiple load and store instructions, automatic state save and restore for interrupts, tightly integrated interrupt controller with Wake-up Interrupt Controller, and multiple core buses capable of simultaneous accesses. Pipeline techniques are employed so that all parts of the processing and memory systems can operate continuously. Typically, while one instruction is being executed, its successor is being decoded, and a third instruction is being fetched from memory.

The LPC1768 is an ARM Cortex-M3 based micro controller for embedded applications requiring a high level of integration and low power dissipation. The ARM Cortex-M3 is a next generation core that offers system enhancements such as modernized debug features and a higher level of support block integration. LPC1768 operate at up to 100 MHz CPU frequency. The ARM Cortex-M3 CPU incorporates a 3-stage pipeline and uses a Harvard architecture with separate local instruction and data buses as well as a third bus for peripherals. The ARM Cortex-M3 CPU also includes an internal pre fetch unit that supports speculative branches.

The peripheral complement of the LPC1768 includes up to 512kB of flash memory, up to 64kB of data memory, Ethernet MAC, a USB interface that can be configured as either Host, Device, or OTG, 8 channel general purpose DMA controller, 4 UART's, 2 CAN channels, 2 SSP controllers, SPI interface, 3 I2C interfaces, 2-input plus 2-output I2S interface, 8 channel 12-bit ADC, 10-bit DAC, motor control PWM, Quadrature Encoder interface, 4 general purpose timers, 6-output general purpose PWM, ultra-low power RTC with separate battery supply, and up to 70 general purpose I/O pins.

2. Technical Specifications

2.1 Specifications of LPC1768:

- ARM Cortex-M3 processor runs up to 100 MHz frequency.
- ARM Cortex-M3 built-in Nested Vectored Interrupt Controller (NVIC).
- Up to 512kB on-chip flash program memory with In-System Programming (ISP) and In-Application Programming (IAP) capabilities. The combination of an enhanced flash memory accelerator and location of the flash memory on the CPU local code/data bus provides high code performance from flash.
- Up to 64kB on-chip SRAM includes:
  - Up to 32kB of SRAM on the CPU with local code/data bus for high-performance CPU access.
  - Up to two 16kB SRAM blocks with separate access paths for higher throughput. These SRAM blocks may be used for Ethernet, USB, and DMA memory, as well as for general purpose instruction and data storage.
- Eight channel General Purpose DMA controller (GPDMA) on the AHB multilayer matrix that can be used with the SSP, I2S, UART, the Analog-to-Digital and Digital-to-Analog converter peripherals, timer match signals, GPIO, and for memory-to-memory transfers.
- Serial interfaces:
  - Ethernet MAC with RMII interface and dedicated DMA controller.
  - USB 2.0 full-speed controller that can be configured for either device, Host, or OTG operation with an on-chip PHY for device and Host functions and a dedicated DMA controller.
  - Four UART's with fractional baud rate generation, internal FIFO, IrDA, and DMA support. One UART has modem control I/O and RS-485/EIA-485 support.
  - Two-channel CAN controller.
  - Two SSP controllers with FIFO and multi-protocol capabilities. The SSP interfaces can be used with the GPDMA controller.
  - SPI controller with synchronous, serial, full duplex communication and programmable data length. SPI is included as a legacy peripheral and can be used instead of SSP0.
  - Three enhanced I2C-bus interfaces, one with an open-drain output supporting the full I2C specification and Fast mode plus with data rates of 1Mbit/s, two with standard port pins. Enhancements include multiple address recognition and monitor mode.
  - I2S (Inter-IC Sound) interface for digital audio input or output, with fractional rate control. The I2S interface can be used with the GPDMA. The I2S interface supports 3-wire data transmit and receive or 4-wire combined transmit and receive connections, as well as master clock output.
- Other peripherals:
  - 70 General Purpose I/O (GPIO) pins with configurable pull-up/down resistors, open drain mode, and repeater mode. All GPIOs are located on an AHB bus for fast access, and support Cortex-M3 bit-banding. GPIOs can be accessed by the

General Purpose DMA Controller. Any pin of ports 0 and 2 can be used to generate an interrupt.

- 12-bit Analog-to-Digital Converter (ADC) with input multiplexing among eight pins, conversion rates up to 200 kHz, and multiple result registers. The 12-bit ADC can be used with the GPDMA controller.
- 10-bit Digital-to-Analog Converter (DAC) with dedicated conversion timer and DMA support.
- Four general purpose timers/counters, with a total of eight capture inputs and ten compare outputs. Each timer block has an external count input. Specific timer events can be selected to generate DMA requests.
- One motor control PWM with support for three-phase motor control.
- Quadrature encoder interface that can monitor one external quadrature encoder.
- One standard PWM/timer block with external count input.
- Real-Time Clock (RTC) with a separate power domain. The RTC is clocked by a dedicated RTC oscillator. The RTC block includes 20 bytes of battery-powered backup registers, allowing system status to be stored when the rest of the chip is powered off. Battery power can be supplied from a standard 3 V Lithium button cell. The RTC will continue working when the battery voltage drops to as low as 2.1 V. An RTC interrupt can wake up the CPU from any reduced power mode.
- Watchdog Timer (WDT). The WDT can be clocked from the internal RC oscillator, the RTC oscillator, or the APB clock.
- Cortex-M3 system tick timer, including an external clock input option.
- Repetitive interrupt timer provides programmable and repeating timed interrupts.

- Standard JTAG test/debug interface as well as Serial Wire Debug and Serial Wire Trace Port options.
- Emulation trace module supports real-time trace.
- Four reduced power modes: Sleep, Deep-sleep, Power-down, and Deep power-down.
- Single 3.3 V power supply (2.4 V to 3.6 V). Temperature range of -40 °C to 85 °C.
- Four external interrupt inputs configurable as edge/level sensitive. All pins on PORT0 and PORT2 can be used as edge sensitive interrupt sources.
- Non-maskable Interrupt (NMI) input.
- Clock output function that can reflect the main oscillator clock, IRC clock, RTC clock, CPU clock, or the USB clock.
- The Wake-up Interrupt Controller (WIC) allows the CPU to automatically wake up from any priority interrupt that can occur while the clocks are stopped in deep sleep, Power-down, and Deep power-down modes
- Processor wake-up from Power-down mode via any interrupt able to operate during Power-down mode (includes external interrupts, RTC interrupt, USB

activity, Ethernet wake-up interrupt, CAN bus activity, PORT0/2 pin interrupt, and NMI).

- Each peripheral has its own clock divider for further power savings.
- Brownout detect with separate threshold for interrupt and forced reset.
- On-chip Power-On Reset (POR).
- On-chip crystal oscillator with an operating range of 1 MHz to 25 MHz.
- 4 MHz internal RC oscillator trimmed to 1% accuracy that can optionally be used as a system clock.
- An on-chip PLL allows CPU operation up to the maximum CPU rate without the need for a high-frequency crystal. May be run from the main oscillator, the internal RC oscillator, or the RTC oscillator.
- A second, dedicated PLL may be used for the USB interface in order to allow added flexibility for the Main PLL settings.
- Versatile pin function selection feature allows many possibilities for using on-chip peripheral functions.

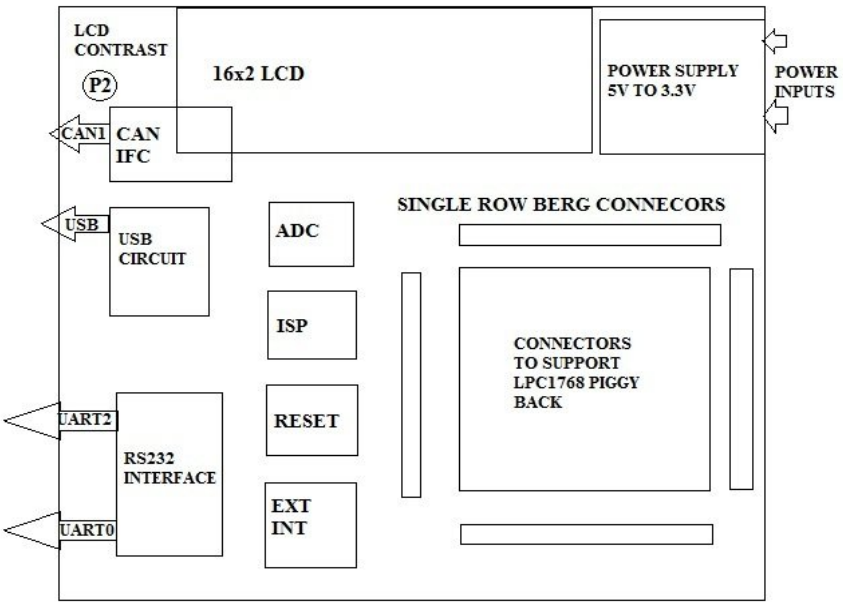
2.2 Specifications of Project Board

- LPC1768 is an ARM Cortex M3 based micro-controller for embedded applications requiring a high level of integration and low power dissipation
- LPC1768 is a 32/16 bit controller with 512 bytes program flash and 64K bytes of RAM
- 12M Hz crystal allows easy communication setup
- On board voltage regulator for generating 3.3V from +5V input through power adapter
- One serial MAX3232 interface with 9 pin DSUB connector for UART0 which is used by the boot loader program, to program LPC178 flash memory without external programmer
- PIGGY BACK Module carrying LPC1768 Controller
- Internal ADC circuit – one preset provided for study
- LCD 16×2 alphanumeric display
- Two general purpose LED's
- Reset switch for resetting the controller
- UART2 provision through 3 pin reliamate
- USB 2.0 device full speed connector, USB link LED
- CAN bus transceiver PCA82C250 with 3 pin reliamate
- Four 20 pin male berg sticks for extending controller signals to any interface board
- 16 no's of flying leads for connecting interface board with project board

- Standard JTAG connector with ARM 2×10 pin layout for programming/ debugging with ARM-JTAG
- 20 pin Standard Trace Port for tracing CPU instruction execution
- A number of software examples in 'C-language' to illustrate the functioning of some of the interfaces. The software examples are compiled using an evaluation version of KEIL4 'C' compiler for ARM
- 5V DC adapter and RS232 cable are provided

3. Hardware Description

3.1 Block Diagram



3.2Connector details:

CN1 to CN4 are the connectors support the mounting of the piggy back board on base board. While inserting the piggy back connectors 1 to 4 of the piggy back board are to be inserted to connectors 1 to 4 of the base board respectively.

CN1 – 28 pin connector 14×2 header

Pin No	Description
1	TDO-JTAG
2	TDI-JTAG
3	TMS-JTAG
4	TRST-JTAG
5	TCK-JTAG
6	P0.26
7	P0.25
8	P0.24
9	P0.23
10, 12	3.3V
11, 15	GROUND
13,14,19,22,23,26,27,28	NC
16	RTCX1

17	RESET
18	RTCX2
20	P1.31
21	P1.30
24	P0.28
25	P0.27

CN2 – 28 pin connector 14×2 header

Pin No	Description
1,2,28	NC
3	P3.26
4	P3.25
5,19	3.3V
6	P0.29
7	P0.30
8,18	GROUND
9	P1.18
10	P1.19
11	P1.20
12	P1.21
13	P1.22
14	P1.23
15	P1.24
16	P1.25
17	P1.26
20	P1.27
21	P1.28
22	P1.29
23	P0.0
24	P0.1
25	P0.10
26	P0.11
27	P2.13

CN3 – 28 pin connector 14×2 header

Pin No	Description
1,2,28	NC
3	P2.12
4	P2.11
5	P2.10
6,23	3.3V
7,24	GROUND
8	P0.22
9	P0.21
10	P0.20
11	P.19
12	P0.15
13	P0.17
14	P0.15
15	P0.16
16	P2.9
17	P2.8
18	P2.7
19	P2.6
20	P2.5
21	P2.4
22	P2.3
25	P2.2
26	P2.1
27	P2.0

CN4 – 28 pin connector 14×2 header

Pin no	Description
1	P0.9
2	P0.8
3	P0.7
4	P0.6
5	P0.5

6	P0.4
7	P4.28
8, 22	GROUND
9, 21	+3.3V
10	P4.29
11	P1.17
12	P1.16
13	P1.15
14	P1.14
15	P1.10
16	P1.9
17	P1.8
18	P1.4
19	P1.1
20	P1.0
23	P0.2
24	P0.3
25	RTCK
26, 27, 28	NC

CN5 to CN8 are single row berg sticks for external use. Available GPIO's are routed to these connectors.

CN5 – 20 pin single row male berg

Pin no	Description
1	P0.26
2	P0.25
3	P0.24
4	P0.23
5	P1.31
6	P1.30
7	P0.28
8	P0.27
9	P3.26
10	P3.25
11	P0.29

12	P0.30
13	P1.18
14	P1.19
15	P1.20
16	P1.21
17	P1.22
18	P1.23
19, 20	Ground

CN6 – 20 pin single row male berg

Pin no	Description
1	P1.24
2	P1.25
3	P1.26
4	P1.27
5	P1.28
6	P1.29
7	P0.0
8	P0.1
9	P0.10
10	P0.11
11	P2.13
12	P2.12
13	P2.11
14	P2.10
15	P0.22
16	P0.21
17	P0.20
18	P0.19
19, 20	Ground

CN7 – 20 pin single row berg

Pin no	Description
1	P0.18
2	P0.17

3	P0.15
4	P0.16
5	P2.8
6	P2.8
7	P2.7
8	P2.6
9	P2.5
10	P2.4
11	P2.3
12	P2.2
13	P2.1
14	P2.0
15	P0.9
16	P0.8
17	P0.7
18	P0.6
19,20	Ground

CN8 – 20 pin single row male berg

Pin no	Description
1	P0.5
2	P0.4
3	P4.28
4	P4.29
5	P1.17
6	P1.16
7	P1.15
8	P1.14
9	P1.10
10	P1.9
11	P1.8
12	P1.4
13	P1.1
14	P1.0

15	P0.2
16	P0.3
17	RTCX1
18	RTCX2
19,20	Ground

CN9 – 20 pin box type FRC male connector. A JTAG interface can be done using this connector.

Pin no	Description
1,2	+3.3V
3	TRST-JTAG
4,6,8,10,12,14,16,18,20	Ground
5	TDI-JTAG
6	TMS-JTAG
7	TCK-JTAG
9	TCK-JTAG
11	RTCK-JTAG
13	TDO-JTAG
15	RESET
17,18	Resistive ground

CN10 – 20 pin box type FRC male connector. Trace port connector.

Pin no	Description
1	+3.3V
2	TMS-JTAG
3,5,7,9,11,13,15,17,19	TCK-JTAG
4	TCK-JTAG
6	TDO-JTAG
8	TDI-JTAG
10	RESET
12	P2.6 – TRACE CLOCK
14	TDO-JTAG/TDATA0
16	TRST-JTAG/TDATA1
18	TDATA2
20	TDATA3

CN11 – USB B type male connector.

Pin no	Description
1	VBUS
2	D-
3	D+
4	Capacitive ground

CN12 – Single row 16 pin female berg – LCD connector.

Pin no	Description
1	Ground
2	+5V
3	LCD contrast
4	RS
5	R/W
6	En
7 to 14	Data 0 to 7
15	Back light anode
16	Back light cathode

CN13 – 9 pin DSUB female connector – UART0 for programming and serial communication.

Pin no	Description
1, 4, 7, 9	NC
2	Receive (Transmit of PC)
3	Transmit (Receive of PC)
5	Ground
6	DTR
8	RTS

PM1 – Male power jack. 1 - +5v. 2, 3 – Ground.
PM2 – Male power mate of 3 pin. 1, 3- Ground. 2 - +5V.
RM1 – Male relimate of 3 pin. UART2 connector. 1 – TXD2, 2 – RXD2, 3 – Ground.
RM2 – Male relimate of 3 pin. CAN1 connector. 1 – CANL1, 2 – CANH1, 3 – Ground.

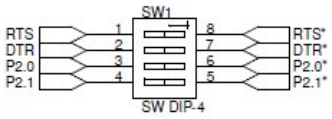
3.3 Jumper Details

Jumper	Description
JP1(1, 2) JP1(2, 3)	Connects RXD0 from U1 to P0.3 of controller Connects CN8 pin 17 to P0.3 of controller
JP2(1, 2)	Connects RXD2 from U1 to P2.9 of controller

JP2(2, 3)	Connects CN7 pin 5 to P2.9 of controller
JP3	Connects ISP control line of the controller (P2.10) to the ISP circuit
JP4	Connects POT1 output to P0.23(AD0.0) pin of controller
JP5	Connects interrupt circuit output to P2.13 (EINT3) pin of controller
JP6	Connects Vbus pins of USB connector (pin 1) and P1.30 of controller
JP7	Connects RTCK-JTAG line to ground via 10k resistor
JP8	Connects output of LM1117 (3.3V) to the board. Can be used to measure current of 3.3V devices
JP9	Can be used to measure the current drawn by 5V devices.
JP10	Connects TXD line of CAN IC to P0.1 of controller
JP11	Connects RXD line of CAN IC to P0.0 of controller

3.4 Switches

SW1 – 4 pin dip switch.



Pins 1 & 8 – RTS: Signal from PC to control P0.14 while programming and RTS*: Signal to ISP circuit.
Pins 2 & 7 – DTR: Signal from PC to control reset while programming. DTR*: Signal to control RESET circuit.
Pins 3 & 6 – P2.0: for GPIO and P2.0*: for GP LED
pins 4 & 5 – P2.1: for GPIO and P2.1*: for GP LED
SW2 – Switches reset pin to ground.
SW3 – Switches EINT3 to ground.

3.5 IC's Detail:

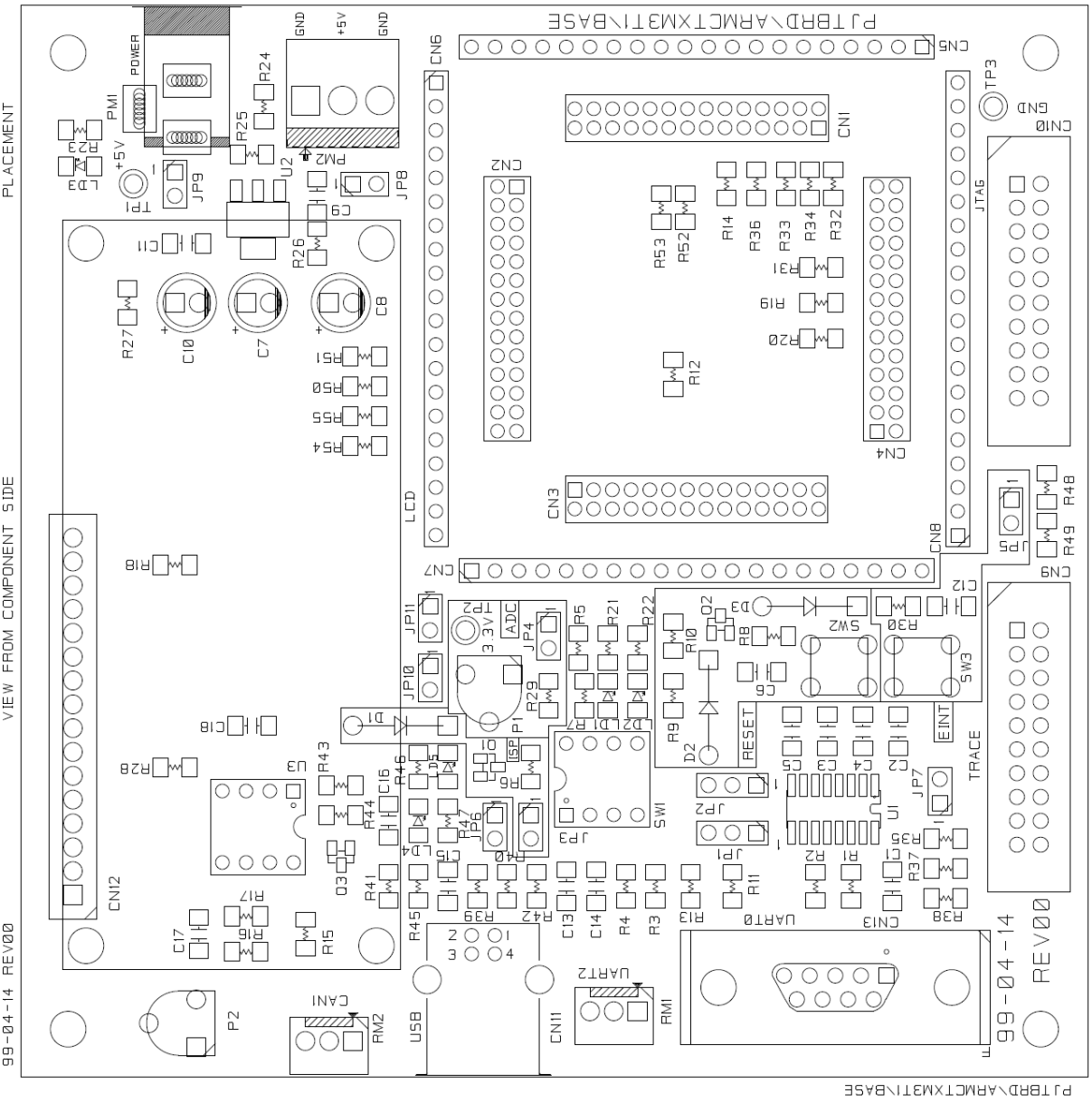
U1 – Max3232: multichannel RS-232 line driver/receiver for serial communication.
U2 – LM1117: An adjustable regulator to generate 3.3V from 5V input.
U3 – PCA82C250: CAN interface IC to interface between controller and physical bus.

3.6 Hardware description of piggy back board

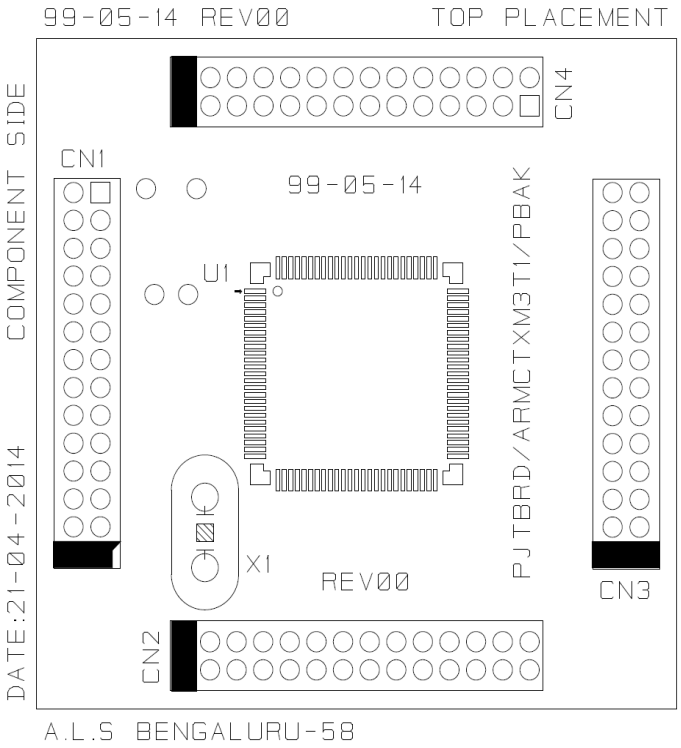
Refer the section 3.2 for the details of CN1 to CN4
X1: 8MHz crystal to runs the controller CPU
X2: 32.768K Hz crystal to run the Internal RTC of the controller
U1: LPC1768FBD100. 100 pin controller IC.

4. Placement Diagram

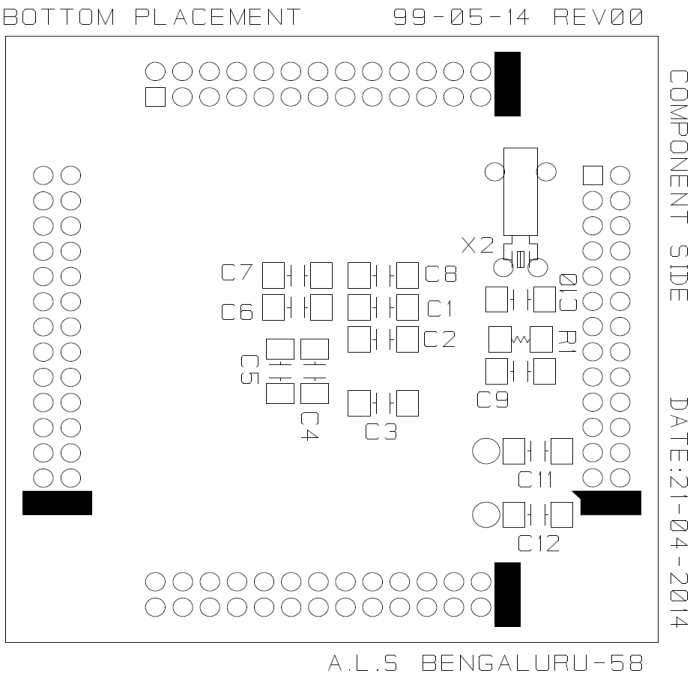
4.1 99-04-14 Base Board



4.2 99-05-14 Piggy back Top view



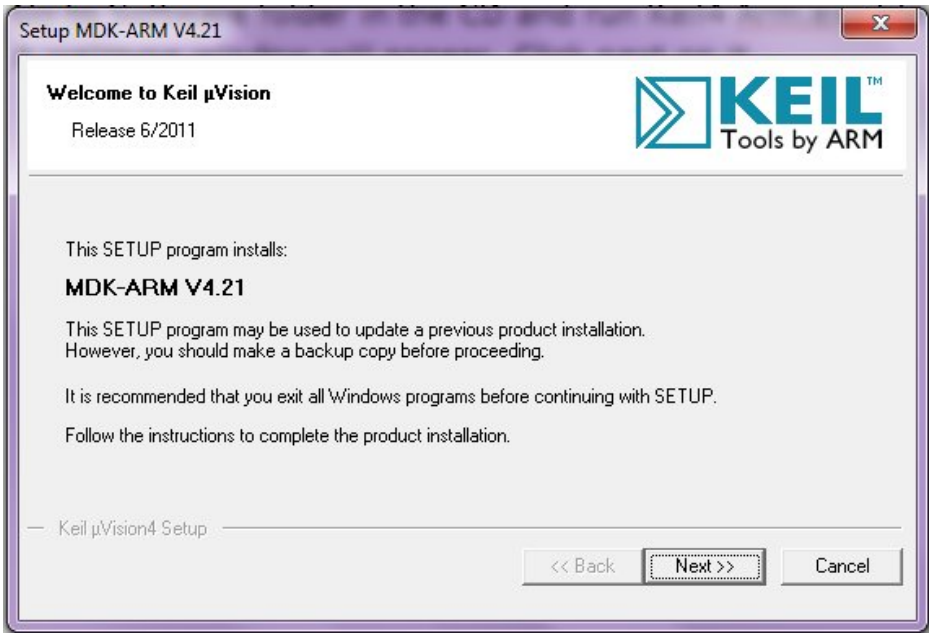
4.3 99-05-14 Piggy back Bottom View



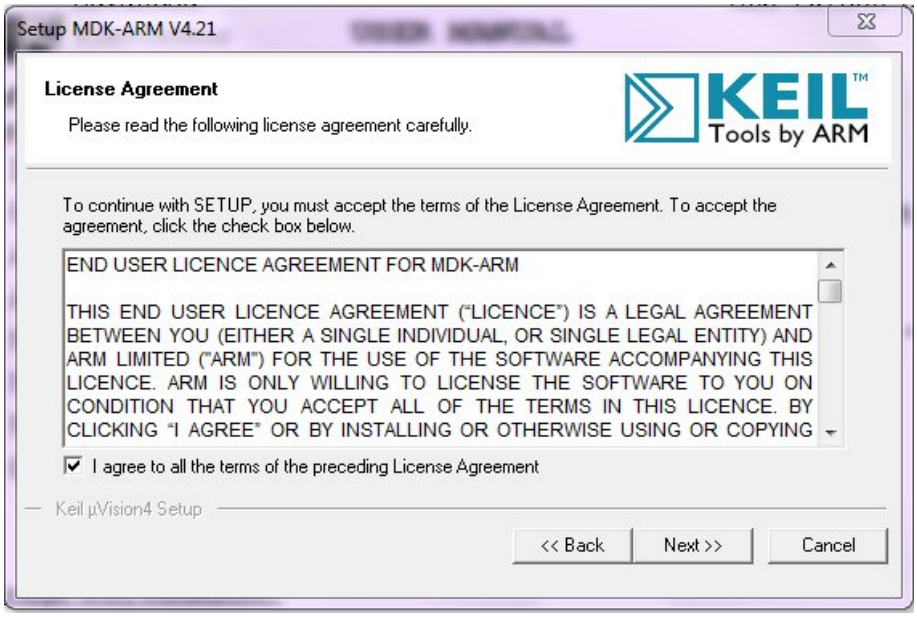
5. Software/Firmware

5.1 Keil uvision4 ide Installation:

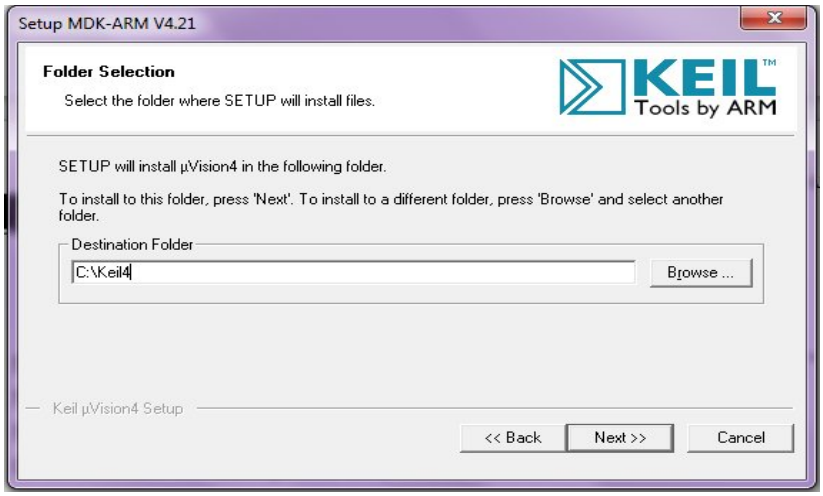
- Installation of keiluVision4 as follows.
- Go to Software folder in the CD and run Keil4 Arm.exe file.
- A welcome window will appear. Click next on it



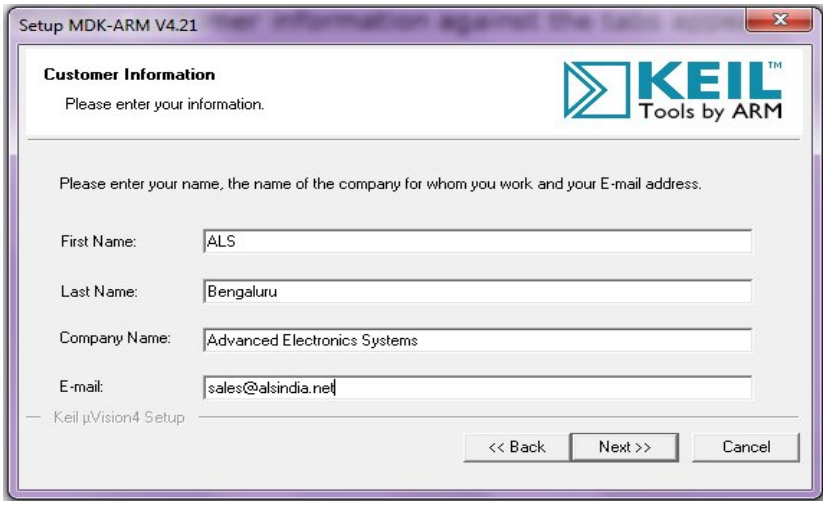
- A license window will appear. Click next



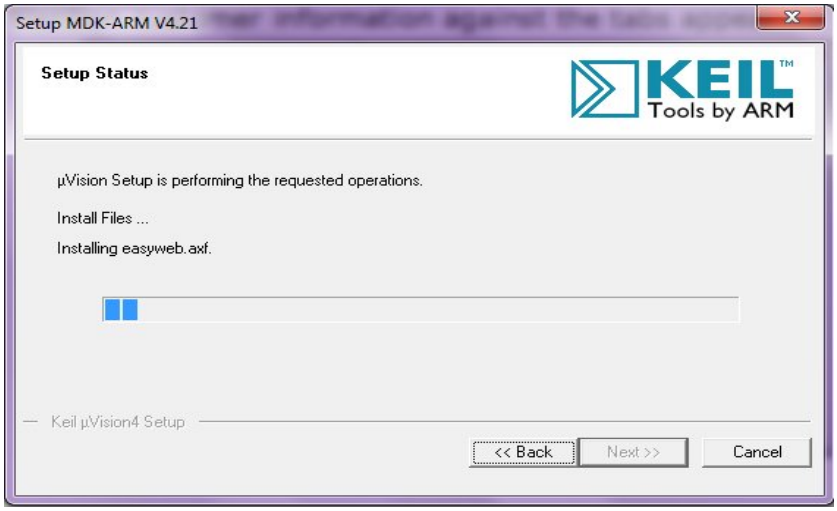
- Choose a folder to install files. Create a separate folder with the name Keil4 at C: drive



- Mention customer information against the tabs appear



- After clicking next a setup status will appear

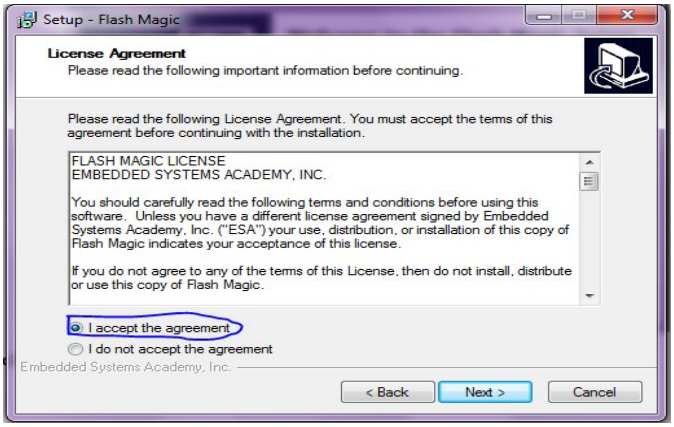


- Click next and finish

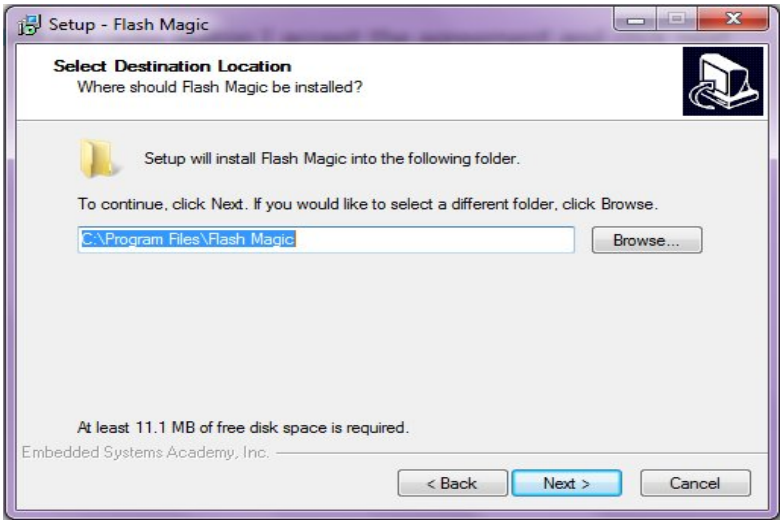
5.2 Flash magic 6.01 installation:

- Go to Software folder in the CD and run FlashMagic.exe file.

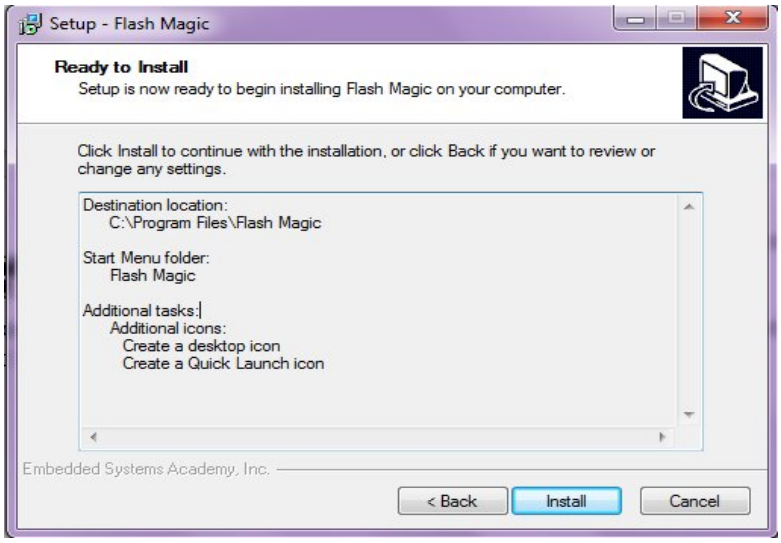
- Click next on Welcome wizard



- Select the radio button I accept the agreement and click next



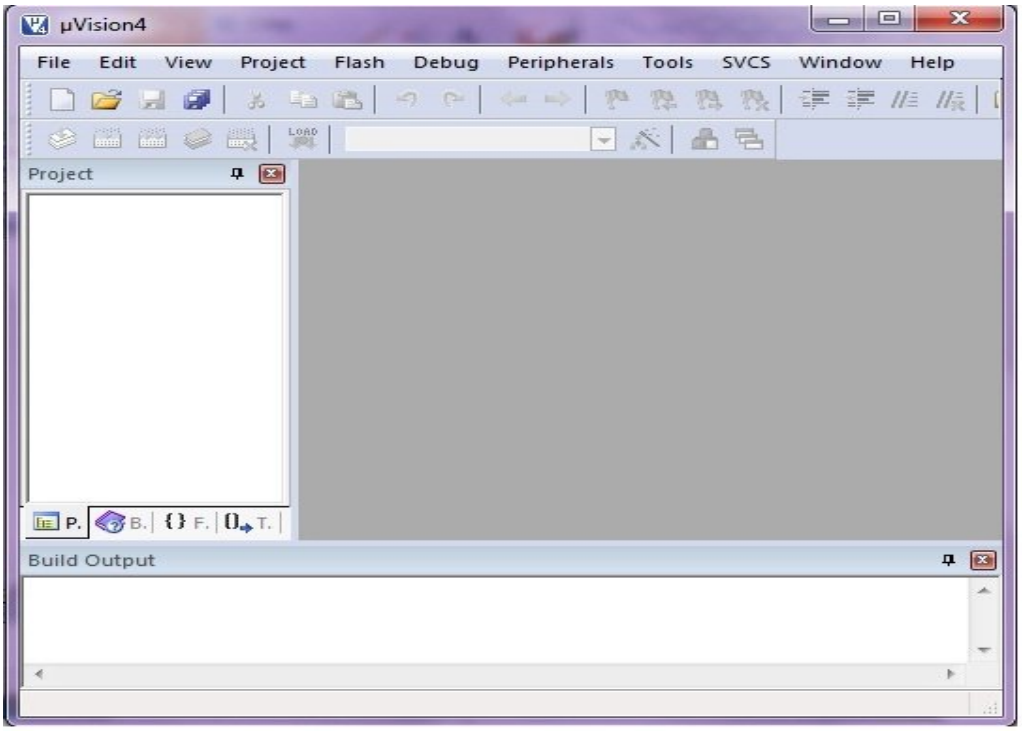
- Choose a folder to install the files. Click next and choose the option create short cut icon and click next. Displays the options we have selected



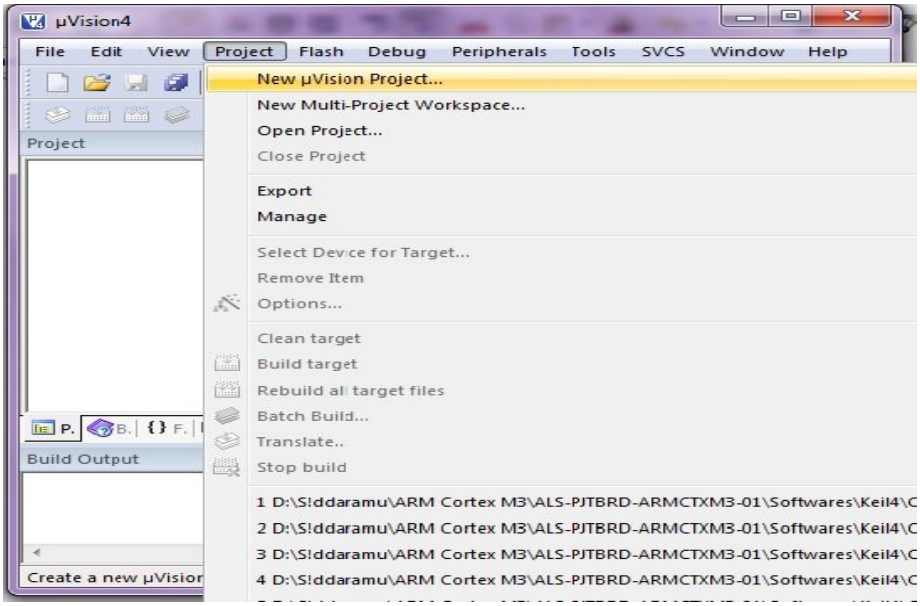
- Click on install and finish

5.3 Project Creation in Keil uvision4 IDE:

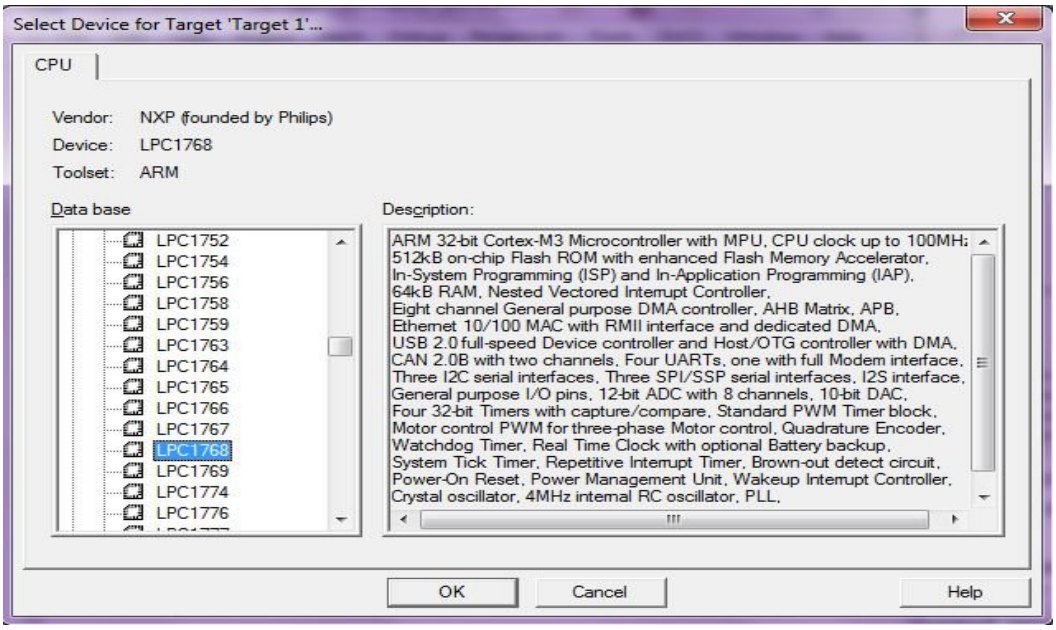
- Create a project folder before creating NEW project.
- Use separate folder for each project
- Open Keil uVision4 IDE software by double clicking on "Keil Uvision4" icon.

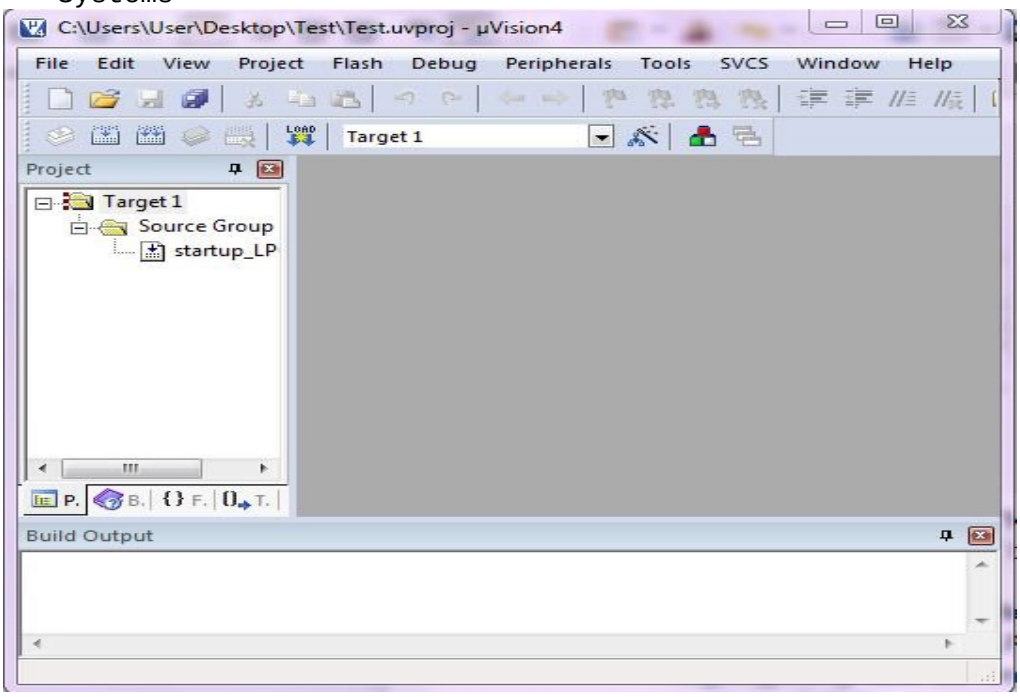


- Go to "Project" then to "New Project" and save it with a name in the Respective Project folder, already you created.

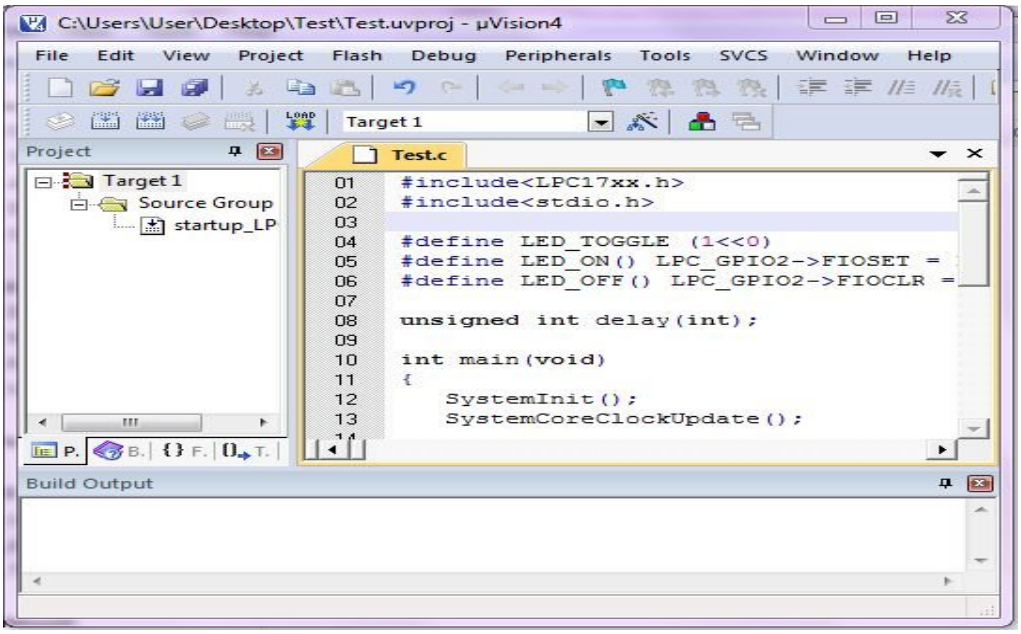


- Select the device as "NXP (founded by Philips)" In that "LPC2148" then Press OK and then press "YES" button to add "startup.s" file.

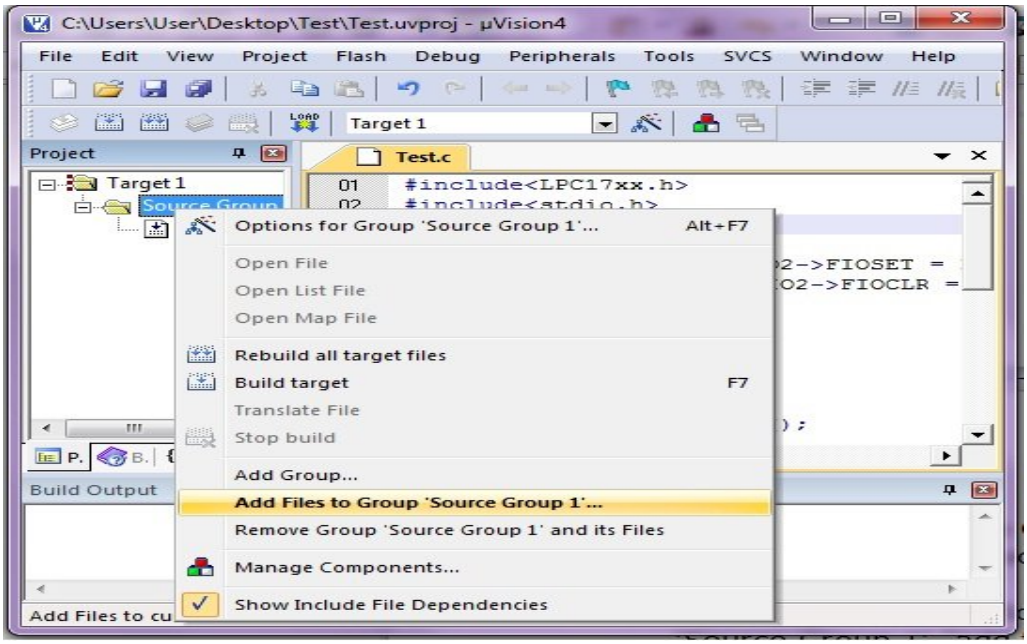




- Go to "File" In that "New" to open an editor window. Create your source file And use the header file "LPC2148.h" in the source file and save the file. Color syntax highlighting will be enabled once the file is saved with a Recognized extension such as ".C ".



- Right click on "Source Group 1" and select the option "Add Files to Group 'Source Group 1' "add the. C source file(s) to the group.



- Again right click on Source Group 1 and select the option "Add Files to Group 'Source Group 1' "add the file – C:Keil(4)\ARM\startup\NXP\LPC17xx\system_LPC17xx.c
  Important: This file should be added at each project creation.
- After adding above file, any changes made to the fill will affect other projects since source file is edited. So to avoid this error, copy the file from source folder and paste to the project folder at each time and add to the project.
- After adding the source file you can see the file in Project Window. Refer section 5.3.1
- Then go to "Project" in that "Translate" to compile the File (s).
- Go to "Project" in that "Build Target" for building all source files such as ".C",".ASM", ".h", files, etc…This will create the .hex file if no warnings & no Errors.

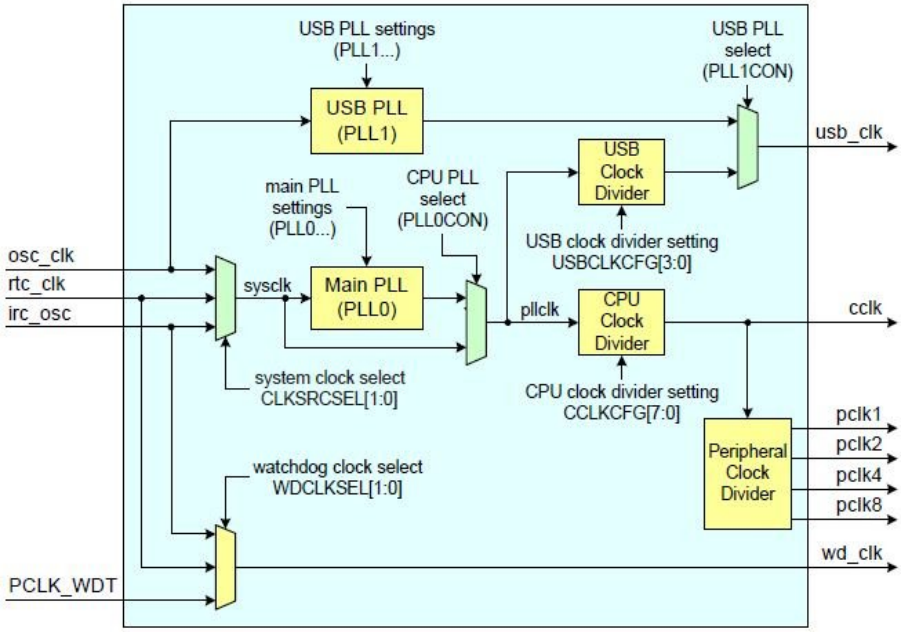
5.3.1 Some Settings to be done in KEILUV4 for Executing C programs:

- In Project Window Right click "TARGET1" and select "options for target 'TARGET1'
- Then go to option "Target" in that
  1. Xtal 12.0MHz
  2. Select IROM1 (starting 0x0 size 0x80000).
  3. Select IRAM1 (starting 0x10000000 size 0x8000).

- Then go to option "Output"
  Select "Create Hex file".
- Then go to option "Linker"
  Select use memory layout from target dialog

5.3.2 Settings to be done at configuration wizard of system_LPC17xx.c file

Before configuring the clock registers study the following block diagram.



There are three clock sources for CPU. Select Oscillator clock out of three. This selection is done by CLKSRCSEL register.

Do not enable PLL0 and PLL1. If we disable the PLL0 System clock will be bypassed directly into CPU clock divider register.

Use CCLKCFG register for choosing the division factor of 4 to get 3M Hz out of 12 M Hz Oscillator frequency.

For any other peripherals use the PCLK same as CCLK.

Follow the below mentioned procedure to do these settings.

- Double click on system_LPC17xx.c file at project window
- Select the configuration wizard at the bottom
- Expand the icons
- Select Clock configuration
- Under System controls and Status registers
  - OSCRANGE: Main Oscillator range select — 1MHz to 20MHz
  - OSCEN: Main oscillator enable — √
- Under Clock source select register (CLKSRCSEL)
  - CLKSRC: PLL clock source selection — Main oscillator
- Disable PLL0 configuration and PLL1 configuration
- Under CPU Clock Configuration register(CCLKCFG)
  - CCLKSEL: Divide value for CPU clock for PLL0 4
- Under USB Clock configuration register (USBCLKCFG)
  - USBSEL: Divide value for USB clock for PLL0 — 4
- Under Peripheral clock selection register 0 (PCLKSEL0) and 1 (PCLKSEL1) select Pclk = Cclk for all.
- Under Power control for peripherals (PCONP)

Enable the power for required peripherals

- If CLKOUT to be studied configure the Clock output configuration register as below
  CLKOUTSEL : Main Oscillator
  CLKOUTDIV : 1
  CLKOUT_EN : √
- Call the functions
  SystemInit();
  SystemCoreClockUpdate();
  at main function without missing. These functions are defined at system_LPC17xx.c where actual clock and other system control registers configuration takes place.
- A small change is required in the file system_LPC17xx.c after installation. Go to text editor:
  #define PLL0_SETUP 0
  #define PLL1_SETUP 0
  if the above #defines are 1 then make 0
- At void SystemInit (void) function, in the condition
  #if (CLOCK_SETUP)
  After the instructions
  LPC_SC->PCLKSEL0 = PCLKSEL0_Val; /* Peripheral Clock Selection */
  LPC_SC->PCLKSEL1 = PCLKSEL1_Val;
  add the below mentioned instruction.
  LPC_SC->CLKSRCSEL = CLKSRCSEL_Val;
  Note: this instruction is written under the condition #if PLL0_SETUP by default.

5.4 Project Downloading in flash magic 6.01:

- Flash magic software can be used to download the HEX files to the Flash Memory of controller.
- To Download Connect the serial cross cable from 9-pin DSUB Female connector of max232 jig to the PC COM port. Switch ON the power.

Some Settings in FLASH MAGIC:

- Step1. Communications:
  Device : LPC1768
  Com Port : COM1
  Baud Rate : 9600
  Interface : None(ISP)
  Oscillator : 12MHz
- Step2. ERASE:
  Select "Erase Blocks Used By Hex File".
- Step3. Hex file:
  Browse and select the Hex file which you want to download.
- Step4. Options:
  Select "Verify After Programming".
- Step5. Load the hex file to be downloaded
- Step5. Start:

  Click Start to download the hex file to the controller.

Note: In FLASH MAGIC Go to Options and then in Advanced Options

- Select High Speed Communications (115200) in Communications tab.
- Select Use DTR & RTS to control RST & ISP Pin and Keep RTS asserted
- while COM Port open.

6. TEST SET UP & TEST PROCEDURE:

6.1 TEST SET UP REQUIREMENTS:

- Cortex M3 project board : 1 No.
- One standard interface board : 1 No.
- Power supply (+5V) : 1 No.
- Cross cable for programming and serial communication :1 No
- Few flying leads
- Kiel uvision4 and flash magic 6.01
- One working COM port (Ex: COM1) in the host computer system and PC for downloading the software.

6.2 TEST SET UP:

- Connect +5V power with respect to GND by using power adapter.
- If you using any interface programs give +5V and GND to interface board from project board.
- Switch on the power supply.
- Before going to do any test, check any IC's are getting heated and also check +5V with respect to GND in board to respective IC’s.
- Also make sure the regulator output which is flowing into the controller

6.3 SAMPLE PROGRAMS:

- Do the connection as mentioned above test set up.
- Switch ON power supply.
- Download the respective source code.hex file using UART0 (DB1). Connect the board to the serial port of a PC using the Max3232 cable provided.

  Note:
- Before downloading the .hex file switch on SW1 (1, 2), short jumper JP3 for ISP. Short JP1(1, 2). After downloaded .hex file switch off SW1 (1, 2), reset the hardware by pressing switch SW2 and observe the respective output.

6.4 Hyper terminal Setup

- Hyper terminal is available as pre installed package at Operating System windows XP.
- Open Hyper terminal Enter the name and choose the icon for the connection. Press OK.
- Enter the details. Choose Com port press OK.
- Press Restore defaults at port settings press ok.
- Press Call → Disconnect.
- File → Properties. Select settings.
- Choose ASCII setup and check the following options.
  Send line ends with line feeds
  Append line feeds to incoming line ends
- Now select Call -> Call.

A few example of on board peripherals given below.

External Interrupt:

Download the "Ext_Int.hex" file. By using P2.12 port line we are generating external interrupt (EINT3). Short JP5, and switch on SW1 pin no.3, When we press the switch SW3 the port line goes low & the external interrupt occurs at port line P2.11.

Led toggle:

Download the "LED_Test.hex" file. General purpose led LD1 is connected to port line P2.0 through SW1 pin number 3, Switch on the pin 3 of SW1 to toggle led.

16X2 LCD:

Download the "LCD_Test.hex" file. A 16X2 Alphanumeric LCD Display with back light is provided along with this Project board and connect LCD to CN12. The message will be displayed on LCD.

For contrast adjust variable pot P2.
RS = 0 for sending Command to the LCD, controlled by port P3.25
RS = 1 for sending Data to the LCD, controlled by port P3.25
R/W = 1 for reading from the LCD, controlled by the port P3.26
R/W = 0 for writing to the LCD, controlled by the port P3.26
EN = 0 for disabling the LCD
EN = 1 for enabling the LCD, controlled by port P4.28
D4 to D7: Ports P1.20 to P1.23.

Test internal ADC:

Download the "Int_ADC.hex" file. This example scans the channel ADC0.0. Voltage at the pin P0.23 is varied by varying the pot P1. Since reference voltage is 3.3V, ADC output range is 000 to FFF (12 bit). Observe the corresponding analog & digital value on LCD. Short jumper JP4.

Test internal RTC:

Download the "Int_RTC.hex" file. Connect the cross cable between DB1 (uart0) and PC com port. Open the hyper terminal. Switch off SW1 (1, 2) and Press the reset switch (SW2) to run the program. According to this software, Internal RTC is operated based on the commands sent from PC terminal through UART0. A read and write operation is done and read values are sent to serial port UART0. Untill Esc key pressed at PC key board a read operation will continue.

A Menu will be displayed on the hyper terminal as follows.

1. RTC WRITE
2. RTC READ
PRESS 1 or 2

If we Press 1 the predefined Values are written to the Real Time Clock (RTC). After the Write RTC operation completes the message will be displayed on the hyper terminal as,

RTC DATA WRITTEN

Then if we press 2 the values written to RTC are read. The o/p of the RTC Read can be seen on hyper terminal as,

```
RTC DATA IS

Year Month     DOY  DOW DOM  HOUR      MIN  SEC
2011 12        365  6    31   23        59   55
```

Test UART0:

Download UART0_Test.hex

The board has an RS-232 serial communication port. The RS-232 transmits and receives signals that appear on the female 9-pin DB connectors (DB1). Use a standard RS-232 cross cable to connect the board to the computer's serial port. The controller provides serial I/O data at TTL levels to the MAX3232 (U3) device, which in turn converts the logic value to the appropriate RS-232 voltage level. In this software a data is received from UART0 and same data is sent back to the same port. Use PC communication terminal to see the working. A character typed at the PC key board is sent to the project board via Hyper terminal. And same character is sent back from the controller to hyper terminal. The data is displayed on Hyper terminal. Connect the cross cable between DB1 (uart0) and PC com port. Open the hyper terminal. Switch off SW1 (1, 2) and Press the reset switch (SW2) to run the program.

Test UART2:

Download UART2_Test.hex

Connect cross cable to reliamate RM1. Pin no. 1 is RXD, 2 is TXD and 3 is GND A key pressed at the key board will get displayed on the hyper terminal via UART1. Type a key at PC key board and check to display the same key on Hyper terminal. Short jumper JP2(1, 2).

To Test Oscillator Clock:

Download CLK_Test.hex. Pin P1.27 of LPC1768 outputs the Clock pulse when it is configured as clock out pin. CLKOUTCFG register allows selecting clock source and division factor. A continuous clock pulse can be observed at the pin P1.27. Observe the 3MHz waveform using CRO at the pin CN6.4.

7. LIST OF DELIVERABLES

PJTBRD\ARMCTXM3T1\BASE	1 No
PJTBRD\ARMCTXM3T1\PBAK	1 No
+5V adapter	1 No
Flying leads	10 No
9 pin cross cable	1 No

8. TROUBLE SHOOTING

Power Supply:

Take care about the +3.3V is properly connected to controller.

In System Programming / Download (ISP):

In System Programming or download could not be established properly then check out whether the following conditions are met

- The cable used for communication should be cross cable and working.
- Switch ON 1 & 2 switches of the SW1 DTR and RTS for serial communication.
- IC MAX3232 is in good condition.
- Jumper JP3 is short

JTAG Programming / Download:

Short jumper JP7 (RTCK) for communication.
Switch off 1 &2 pins of the SW1.

General Problems:

- Make Proper Jumper Connections as mentioned in Hardware Details.
- Make Proper Connections as mentioned in Demo Programs Setup.

1.