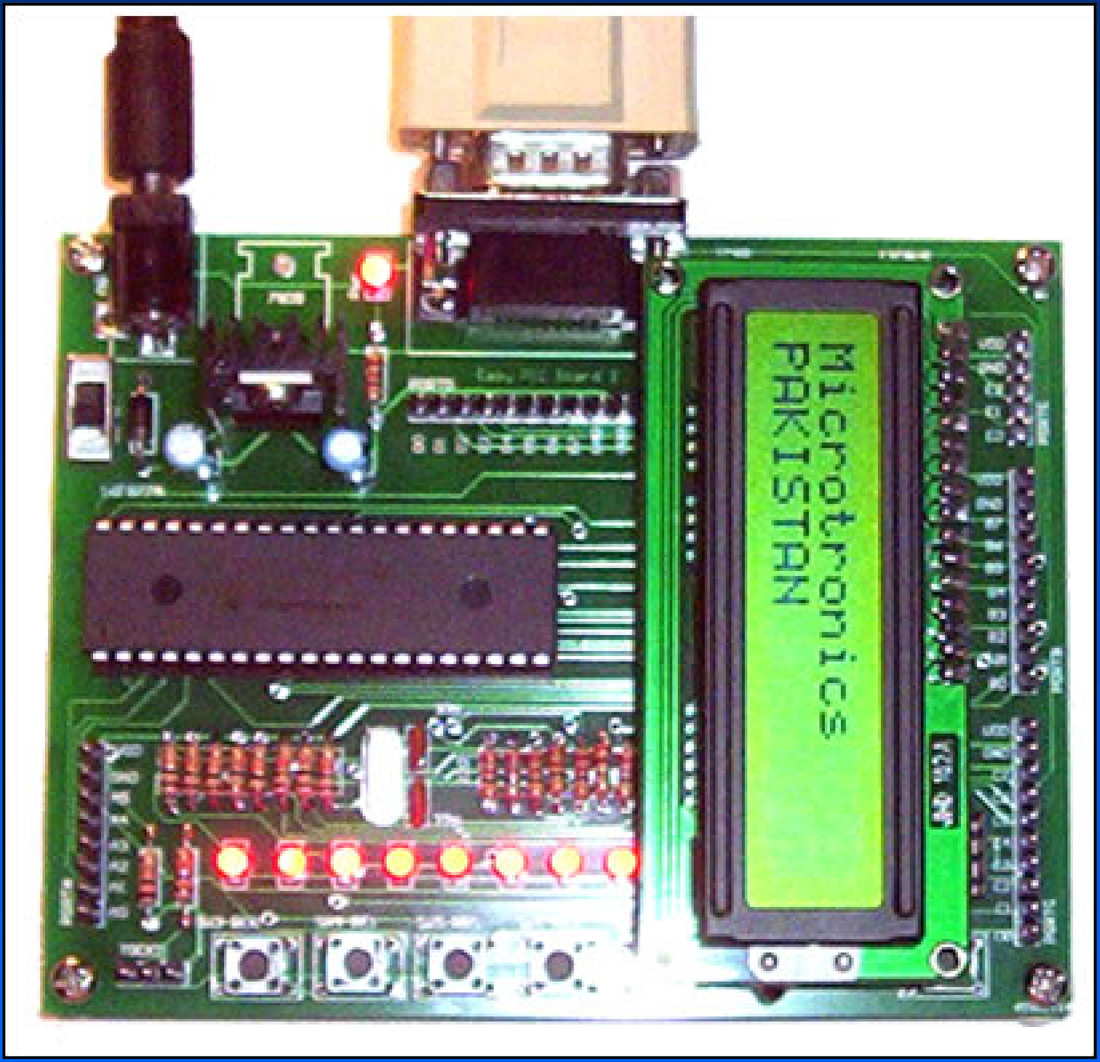
PIC Lab-II Motherboard



User Manual

Congratulations on purchase of PIC Lab-II development board. The board supports most commonly used 40 Pin Mid range PIC microcontroller units, and has on-board many commonly used peripheral devices. All I/O lines are available through separate, well marked headers for daughter boards, or projects. Every I/O line group is provided with a set of power supply, so that the daughter board can be powered up from the motherboard.

Salient Features of Board.

- Socket for 40 Pin PIC Microcontrollers (16F87x, 18FXXX)
- Changeable Crystal Oscillator (20MHz standard)
- 5 Push Switches
- 1 16x2 Character LCD
- USART (RS-232 Module)
- I2C EEPROM
- 1 PIZO Buzzer
- In Circuit Serial programming / Debugging Support
- 8 LED indicators
- 38KHz Modulated Infra Red Sensor
- Connector for two Analog Inputs
- Connector for Hardware driven PWM
- Connector for T0CKI
- Connector for I2C Bus
- All I/O lines available through Headers
- On Board regulated 5V 1A power supply.

Power Supply

The board uses 5V regulated power supply, in order to assure a constant supply of 5V, we recommend a power source more than 5V. The recommendation is 9V at least 300mA load. If you are going to drive an external load through the board, like motor, or LEDs etc, make sure that the supply current is appropriate. In that case we recommend 9V-12V DC adapter, 800mA to 1000mA.

The adapter must have center pin supplying Positive. Reverse polarity is checked however. There is a blocking diode, which will not allow the reverse polarity to reach the board. There is an On/OFF switch, and a Power-ON LED indicator. The Voltage regulator IC, 7805, is rated as 1A, and gets heat up if heavy load is put on it, or if a voltage adapter of more than 9V is used. The regulated is fitted with commercial heat sink.

The Microcontroller

The board contains a socket for 40-Pin PIC microcontroller. Board comes with standard PIC 16F877A microcontroller. However if you want to change, lets say to 18F452, simply take the IC out of socket and insert the one of your choice.

The Oscillator Circuit

The oscillator circuit consists of a crystal oscillator and two 22pF capacitors. The crystal oscillator is not fixed soldered, rather there is a base, into which it can be inserted. The board comes with 20MHz crystal, however if you want to change this to another frequency, just pull it out and insert the new one.

PIC 18F series are capable of multiplying the clock frequency by 4, to generate an internal high frequency oscillator. They can run at a maximum of 40MHz, therefore using a 10MHz crystal, along with enabling an internal PLL, will give a speed of 40MHz.

In Circuit Serial Programming

Traditionally microcontrollers are programmed, using a device called programmer. The microcontroller is taken out of the circuit, and put into the programmer to get it programmed. The processor is then inserted back into the board to run the program. This is however quite cumbersome, to repeatedly take out the processor. Microchip therefore introduced the idea of In Circuit Programming. This is also called ICSP. This board supports In circuit Serial programming. However not all programmers have ICSP support. So you will need a programmer with ICSP support, such as Microtronics PIC-PG2.

ICSP uses two of the microcontroller pins, RB6 and RB7, Called PGM and PGD respectively. If you are going to use ICSP, these pins must not be tied to any circuit that should drain the signals from programmer on these pins. The board, has kept this feature and RB6 and RB7 are not connected to any device. However since these pins are available through headers, you can use them in your applications, then if the ICSP fails, disconnect the device before programming.

The programmer also supplies 5V output, but this power is insufficient to supply the entire board. It is therefore mandatory to **keep the power on while programming** . As soon as you connect the programmer cable to the ICSP connector, the MCLR pin goes low, this inhibits the microcontroller to run any program. Therefore, if your programmer does not have the mechanism to release MCLR pin, then after programming, remove the cable, so that the program can run.

DIP Switch

There is an 8 switch DIP switch, labeled as Enable on board. This switch will connect or disconnect various peripheral devices on board, so that the associated pins can be freed for use in other projects. We will talk about the related Switch, in appropriate topics.

LED Indicators (PORTC.0 … PORTC.7)

PIC Lab-II has 8 Led indicators connected to PORTC of the microcontroller. The LEDs are connected through 220 ohms current limiting resistor, and are sourced by the microcontroller. Therefore a logical 1, or on state of the microcontroller pin will turn the corresponding LED ON.

DIP Switch 1, will enable or disable the LED module.

A number of other devices also use PORTC, therefore simultaneous use or connection of port pins to LEDs may interfere with other device function. For example, USART uses RC5 and RC6 for communication, if LEDs are enabled, this communication may Fail. Same is the case with Boot-Loader programming which uses USART, therefore before programming using Boot-Loader, disable the LEDs using Dip-Switch 1.

Character LCD (PORTD.2,3,4,5,6,7)

The board contains a socket for HD44780 based character LCDs with back light. The LCD module can be any, based upon HD44780 controller. Commonly used is 16x2 character LCD. The module uses 4 bit mode, in which the highest 4 bits of PORTD are connected to data pins of LCD. PORTD.2 is connected to Enable Pin, and PORTD.3 to RS-Pin of LCD. Thus following declarations must be used before initializing the LCD (Proton Basic).

```
LCD_DTPIN PORTD.4
LCD_RSPIN PORTD.3
LCD_ENPIN PORTD.2
```

LCD can be enabled or disabled by using DIP-Switch-2

Infra-Red Remote Control Sensor (PORTA.3)

A large number of devices are now using infra-red remote controls to communicate. The standard protocol is based upon 38KHz, modulated infra-red communication. The PIC Lab-II board contains a 38KHz, modulated Infra red sensor, that will detect the Infra-reds, which are modulated only at 38Khz. Most commonly available remote controls for TV, CD and A/C use the same protocol. However they differ in codes they transmit. The Infra-red Sensor is connected to, PORTA.3. this port is also analog input, therefore make sure it is set to digital, and configured as Input to read the sensor.

The infrared –Sensor can be enabled or disabled using DIP-Switch 3.

Push Switches

There are five push switches available on board. These switches have all 10K pull up resistors and will be active low. Therefore when switch is not pressed the associated microcontroller pin will read a logical high or 1. when switch is pressed, it connects the pin to ground thereby giving a Low state or logical 0.

The switches are named as SW3—SW7

The PORT Pins connected to these switches is as below:

Switch	PORT Pin	Comments
SW3	PORTE.0	DIP-Switch 4 to enable / disable
SW4	PORTE.1	DIP-Switch 5 to enable / disable
SW5	PORTB.0	PORTB.0 is also external interrupt, DIP SW-6
SW6	PORTE.2	Dip Switch 7 to enable / disable
SW7	PORTA.4	PORTA.4 is also external clock input for timer 0. DOP Switch SW 8 to enable/disable

Reset Switch (MCLR)

Reset switch is located on the right hand corner and labeled as RST. When pressed the switch gives a logical 0 to MCLR pin of microcontroller, which is reset, and starts the software from beginning. All RAM data is cleared. EEPROM is preserved. Timers are cleared. The Reset switch is specially useful, for boot-loader programming.

I2C EEPROM (PORTC.3, PORTC.4)

I2C communication bus is very commonly used in electronics devices. More and more devices are coming up with I2C protocol support. The board contains an EEPROM Chip, (24c08) which is 8K EEPROM. This chip is attached to microcontroller using PORTC.3 and PORTC.4. The 40 pin, PICs have these pins attached to internal hardware of I2C communication. PORTC.3 is SCL and PORTC.4 is SDA. Due to built in I2C communication hardware, the software overhead is very much reduced.

The I2C bus contains 10K pull-ups on both SDA and SCL pins.

EEPROM is placed on an 8 pin base, it can be removed and another one inserted, if more memory is required. The EEPROMs have three address pins, which can be used to connect many more EEPROMs in parallel. The board has however given this a permanent address of 000 . The I2C code for EEPROM is 1010. next three bits would be EEPROM chip address, and last bit is direction. Therefore the complete address to write on EEPROM would be: 1010 000 0 and to read it would be : 1010 000 1

I2C BUS (Connector)

The board also provides an I2C Bus as a connector, which contains connection for SDA and SCL pins along with Power for daughter boards. The bus connects directly to PORTC.3 and PORTC.4 which are hardware driven I2C pins. Keep it in mind that these pins have associated 10K pull up resistors already attached. In case these pins are being used for some other application this fact must be kept in mind.

USART (PORTC.6, PORTC.7)

The board contains a standard universal Serial Asynchronous receiver and transmitter. Many devices use this protocol to communicate with other devices. The communication is hardware independent, and just needs two wires, on for transmission and one for receiving data. PCs and some other devices, use a level translator, to redefine the standard signals for logical 0 and 1. this is done so, to minimize noise interference as well as prolong communication distance. To use these signals, they must be converted back to TTL level logic. The PIC-Lab-II board contains Rs-232 level converter which converts these signals to TTL level, and to transmission levels while sending data. Most PIC microcontrollers contain an internal hardware to manage this communication, so that software development becomes easy. PORTC.6 and PORTC.7 are configured as hardware USART communication pins.

NOTE: since the PORTC is also connected to LEDs, if LEDs are enabled receiving data from USART is interfered. It is therefore mandatory to disable, LEDs while using UASRT.

PIZO Buzzer (PORTA.5)

The board contains a connector for PIZO. The PIZO buzzer, module consists of a transistor and two resistors. The transistor connects directly to PORTA.5. The buzzer has to be given oscillatory signals, like a train of 0s and 1s to make a sound.

I/O PORTS

All I/O lines, are available as headers on board for connection to other devices or daughter boards. The headers are separated fro each PORT, and the number of header pins are according to the I/O lines available on the respective port. Each port header is provided with power supply for the daughter board.

T0CKI Connector (PORTA.4)

Although this pin is available trough standard PORTA header, it has been provided separately along with power supply, to used as external clock signals for Timer 0. The pin is also general purpose and can be used as a serial communication between two boards directly connected together. Remember PORTA.4 is open collector and it will need an external PULL up resistor. Fortunately SW7 is also connected to this port pin, and has a pull-up resistor, so enabling SW7 will provide the necessary Pull-Up resistor, in that case don’t press the button, data will be received through TOCKI connector.

Analog input Connectors (PORTA.0, PORTA.1)

Many PIC microcontrollers, give analog to digital conversion on chip. This is quite advantageous, as many sensors give analog data. Although the number of analog inputs on a chip vary, yet if we talk about 16F877A, it has analog inputs on PORTA and E. all these pins are available for projects through PORT headers. However we have specially included two connectors, from PORTA.0 and PORTA.1 along with power supply, on board. These three pin connectors can be used directly to any analog device, giving upto 5V analog signal.

Pulse Width Modulation Connector (PORTC.2)

Pulse Width modulation is a common technique used to alter the speed of motors etc. although PWM can be achieved on any digital pin, PIC microcontrollers have inbuilt hardware, that once set keeps on sending PWM waveform to the device, despite processor busy in other jobs. This is really a multi tasking. 16F877 has two hardware PWM modules, one of them is on PORTC.2 which is available through port header, as well as through this PWM connector, which also has power supply for daughter board if required.

Boot Loader

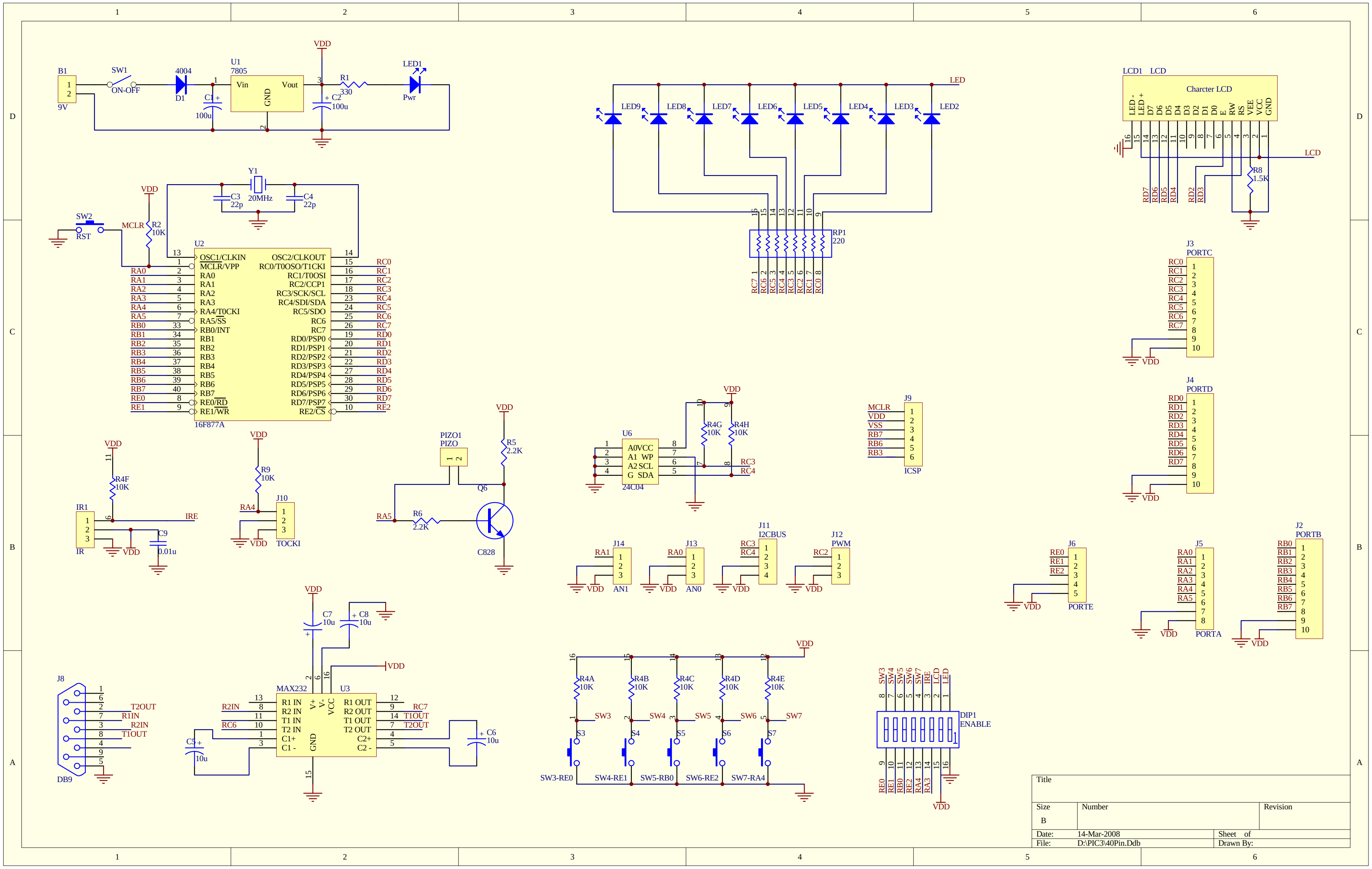
The supplied 16F877A microcontroller comes with a pre-programmed Boot loader software. This software eliminates the need for an external programmer device. The boot loader program has an associated part which will run on windows terminal. The boot-loader is configured to work with 20MHz crystal, and at top BAUD speed of 115K asynchronous serial communication. Therefore configure your windows terminal to use 115K speed.

Load the program hex file, which you want to load. Connect serial cable to PIC Lab-II board. Turn power ON, press reset button, and immediately click on Write button on boot loader windows application. This will immediately write the software into your microcontroller.

External programmer:

In case you want to use an external programmer, select one which has support for ICSP, this will facilitate the programming job, and you will not have to remove the microcontroller from its socket again and again. The board supports even low voltage programming, PGM pin is connected to PORTB.3 When High the microcontroller will be programmed by LVP. This is usually all set by the programmer itself. All you need to know is that the programmer might use RB3, RB6 and RB7 so, your application must not be using them at-least at the time of programming, the pins should be free. By default these pins are free on board. If you are connecting an external board, just remove the connector.

Keep the system power on while programming, as the programmers can give only small amount of power, which is enough for microcontroller, but not for the entire board.


B1
9V
SW1
ON-OFF
4004
D1
U1
7805
Vin
Vout
GND
C1
100u
C2
100u
VDD
R1
330
LED1
Pwr
LED
LED9
LED8
LED7
LED6
LED5
LED4
LED3
LED2
RP1
220
RC7
RC6
RC5
RC4
RC3
RC2
RC1
RC0
LCD1
LCD
Charcter LCD
LED -
LED +
D7
D6
D5
D4
D3
D2
D1
D0
E
RW
RS
VEE
VCC
GND
R8
1.5K
RD7
RD6
RD5
RD4
RD2
RD3
Y1
20MHz
C3
22p
C4
22p
SW2
RST
MCLR
R2
10K
U2
16F877A
OSC1/CLKIN
MCLR/VPP
RA0
RA1
RA2
RA3
RA4/T0CKI
RA5/SS
RB0/INT
RB1
RB2
RB3
RB4
RB5
RB6
RB7
RE0/RD
RE1/WR
OSC2/CLKOUT
RC0/T0OSO/T1CKI
RC1/T0OSI
RC2/CCP1
RC3/SCK/SCL
RC4/SDI/SDA
RC5/SDO
RC6
RC7
RD0/PSP0
RD1/PSP1
RD2/PSP2
RD3/PSP3
RD4/PSP4
RD5/PSP5
RD6/PSP6
RD7/PSP7
RE2/CS
J3
PORTC
J4
PORTD
U6
24C04
A0VCC
A1 WP
A2 SCL
G SDA
R4G
10K
R4H
10K
J9
ICSP
MCLR
VDD
VSS
RB7
RB6
RB3
R4F
10K
IR1
IR
IRE
C9
0.01u
R9
10K
J10
TOCKI
RA4
PIZO1
PIZO
R5
2.2K
Q6
C828
RA5
R6
2.2K
J14
AN1
J13
AN0
J11
I2CBUS
J12
PWM
J6
PORTE
J5
PORTA
J2
PORTB
J8
DB9
T2OUT
R1IN
R2IN
T1OUT
C7
10u
C8
10u
C5
10u
MAX232
U3
R1 IN
R2 IN
T1 IN
T2 IN
C1+
C1 -
V+
V-
VCC
GND
R1 OUT
R2 OUT
T1 OUT
T2 OUT
C2+
C2 -
C6
10u
R4A
10K
R4B
10K
R4C
10K
R4D
10K
R4E
10K
SW3
SW4
SW5
SW6
SW7
S3
S4
S5
S6
S7
SW3-RE0
SW4-RE1
SW5-RB0
SW6-RE2
SW7-RA4
DIP1
ENABLE
IRE
LCD
LED
RE0
RE1
RB0
RE2
RA4
RA3
Title
Size
B
Number
Revision
Date:
14-Mar-2008
Sheet of
File:
D:\PIC3\40Pin.Ddb
Drawn By: