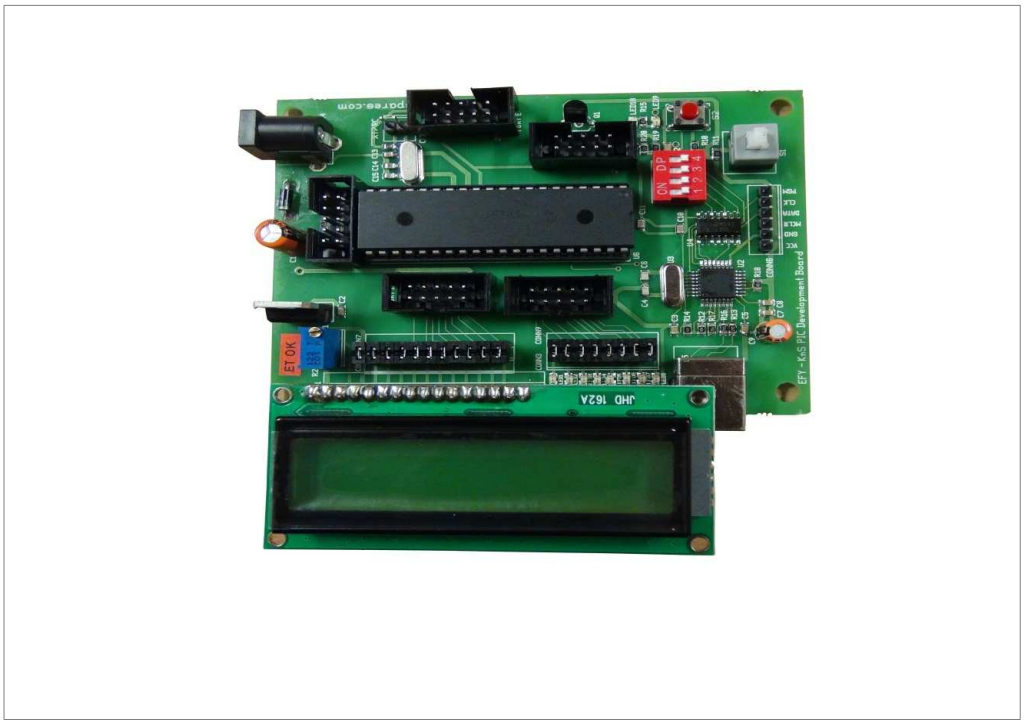
Manual

Kitsnspares.com



[PIC DEVELOPMENT BOARD]

MANUAL

The document describes how to use a PIC Development Board to create applications based on PIC microcontrollers.

Introduction

A PIC Development Board allows you to perform any experiment on PIC microcontrollers and create any applications based on them with minimum amount of effort and time. It's easy and fast programming helps debug the programs at a much faster rate reducing the time to market, whereas on-board interfaces like LCD and LEDs make it really comfortable to test the programs on the go. Connection of any external hardware is also very handy with I/O extended through box headers. Programing of the microcontrollers on the board is done directly through USB cable. The board can also be used as a programmer for programming PIC microcontrollers on the end product.

Features of a PIC Development Board

- All I/Os are extended through box headers to comfortably connect external hardware
- On-board interfaces for LEDs and LCD
- Easy programming through 'B-type' USB
- Can be used as a PIC programmer for programming chip on the end product itself
- XT/RC mode selection facility
- Compact and low cost

Board layout

The layout of a PIC Development Board is shown in Fig.1, indicating the positions of various interfaces on the board.

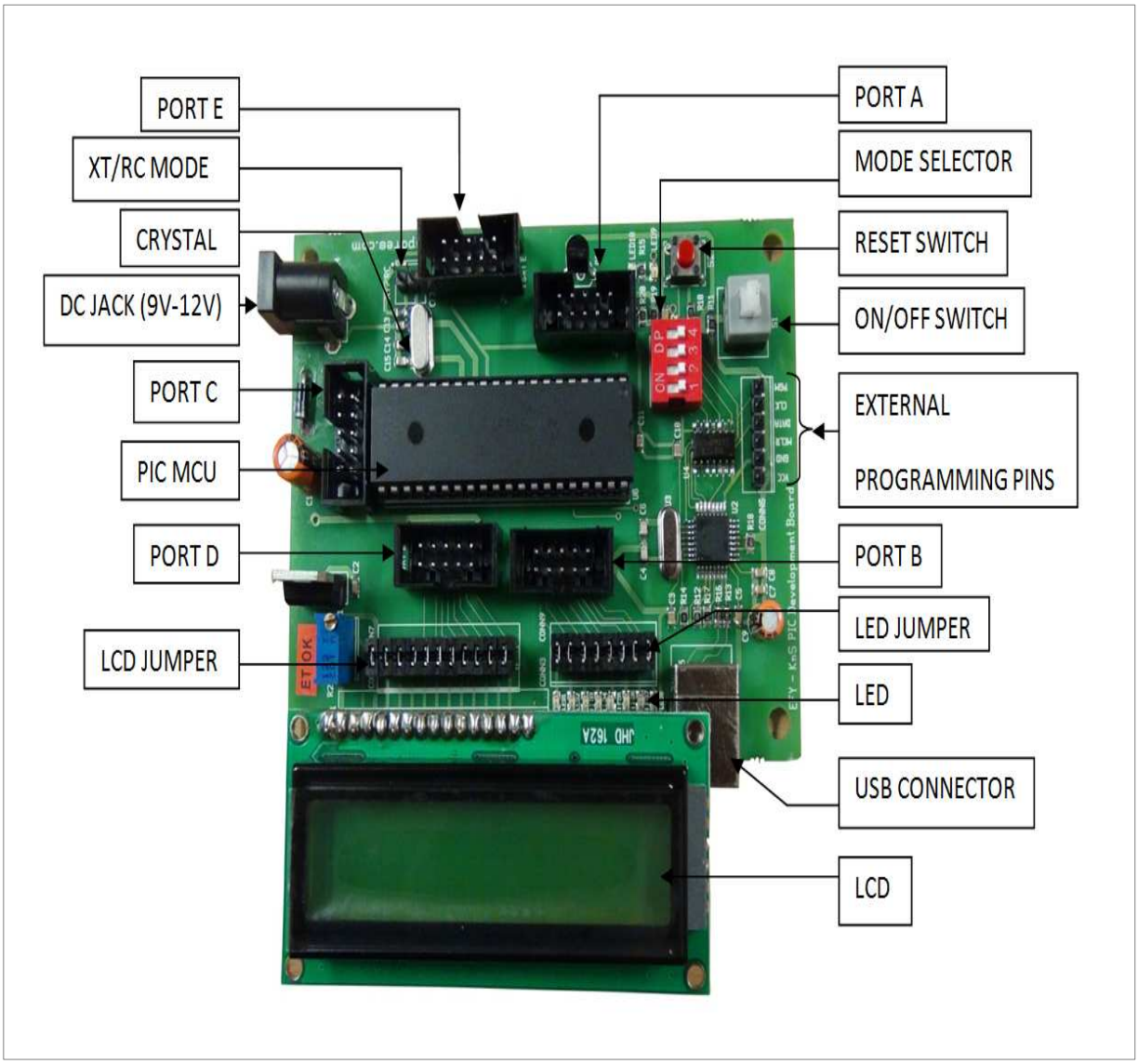


Fig. 1: PIC Development Board layout

Pin diagram

Fig. 2 shows the pin diagram for interfacing external hardware to the board and to use the on-board interfaces for debugging the programs.

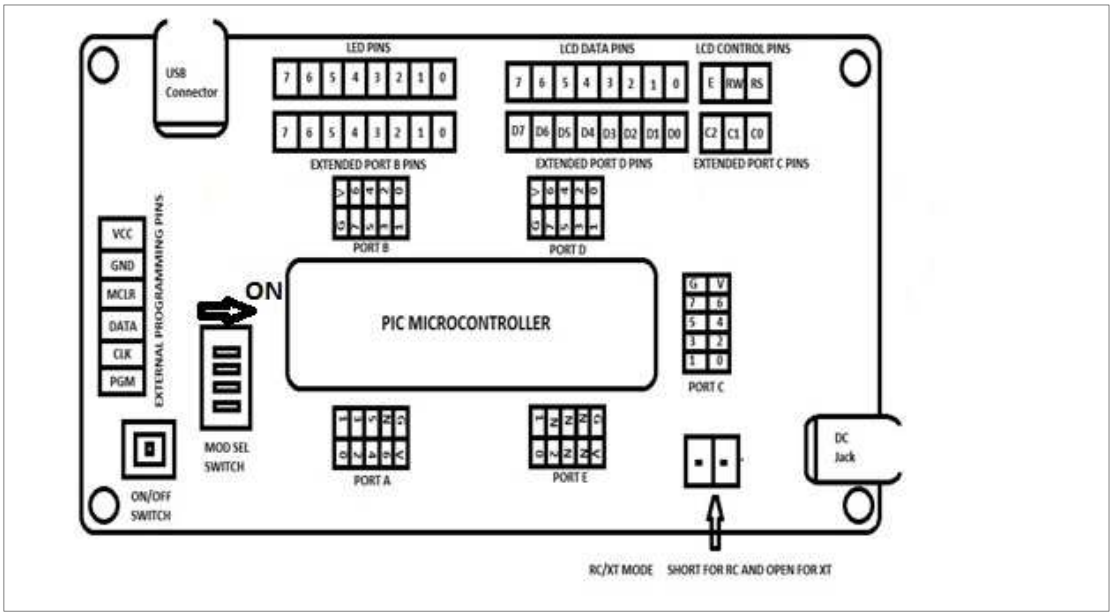


Fig. 2: Pin diagram

N – Not connected

V – Vcc

G - Gnd

Interfaces

1. **I/O:** There are 5 input/output ports that are extended through box headers for easy interfacing to the external hardware using band-cable

   Each box header has pin 9 as Vcc and pin 10 as Gnd (Ground) for supplying power to the external hardware. Port pins are numbered as shown below in Fig. 3.

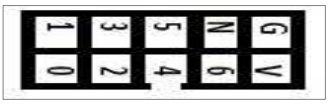


Fig. 3: Box header

2.**LED:** LEDs are provided on the board for quickly debugging simple programs.

These can be connected to Port B using a mini jumper.

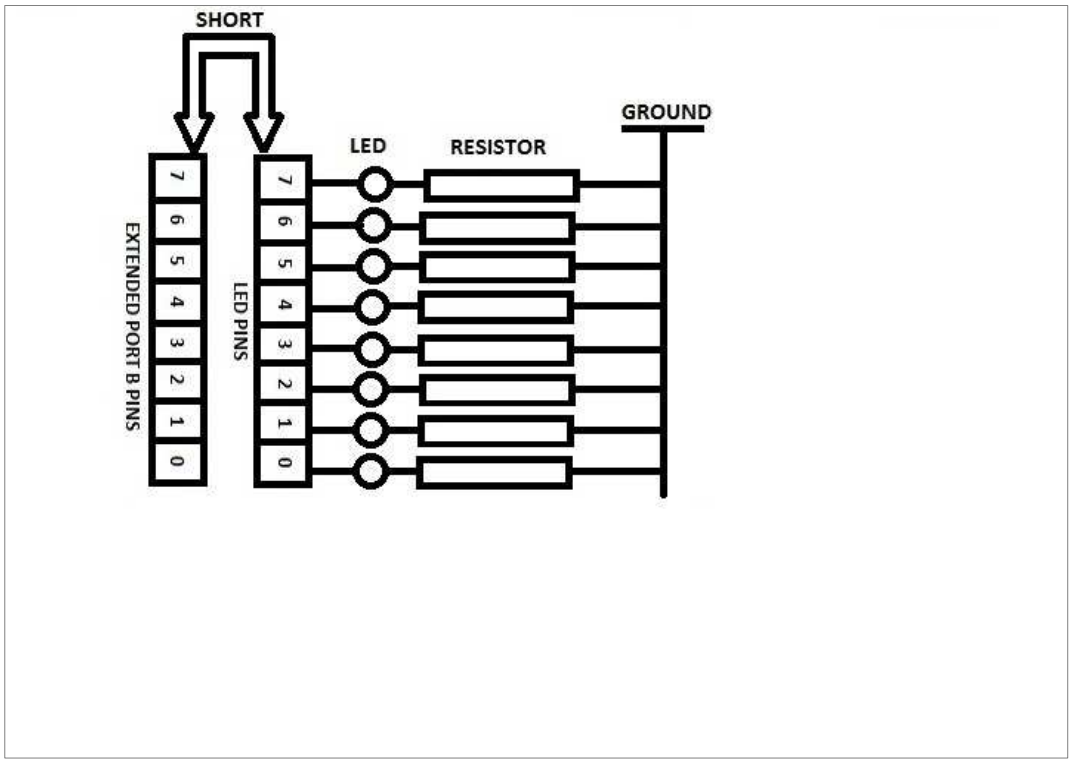


Fig. 4: LED interfaces

3.**LCD:** LCD provided in the board can be connected to Port C and Port D. Port C's initial 3 pins are used as the control signals, while the complete Port D is used as the LCD data pins. To connect the LCD to the PIC microcontroller ports, you need to short the two connectors using mini jumpers provided with the kit.

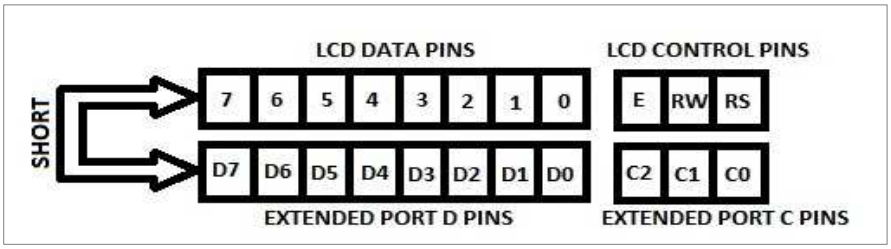


Fig. 5: LCD interface

4. **External programmer:** The external programming pins can be used to program any PIC microcontroller on some external hardware.

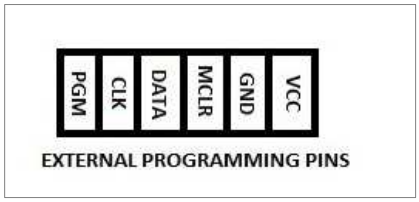


Fig. 6: External programmer

Installation of USB to Serial drivers

To use a PIC Development Board you need to install USB to Serial converter drivers in your computer.

If the computer doesn't have the USB to Serial drivers, the moment you will plug in the device to it through the USB port, it would display the following message shown in the Fig.7.

Fig. 7: Hardware detect

To install USB to Serial drivers, open device manager and double click the option USB<->Serial as shown in the Fig. 8.

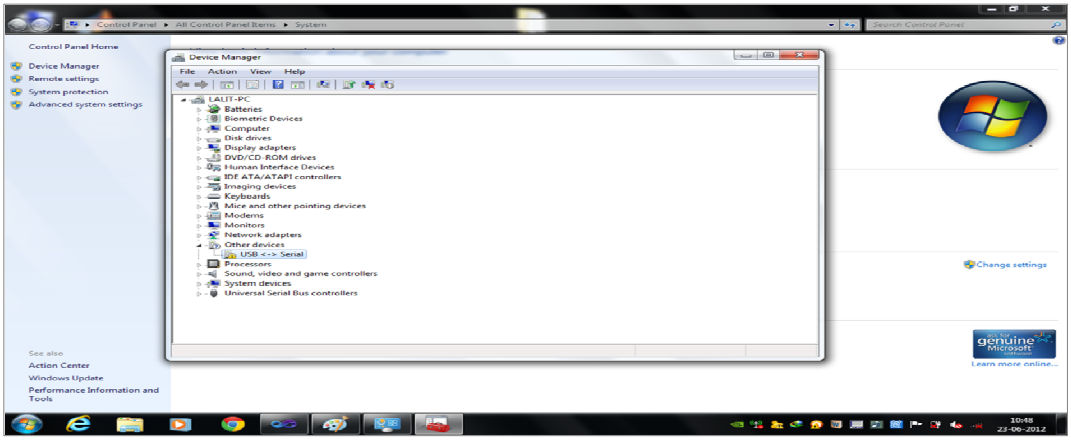


Fig. 8: Device manager

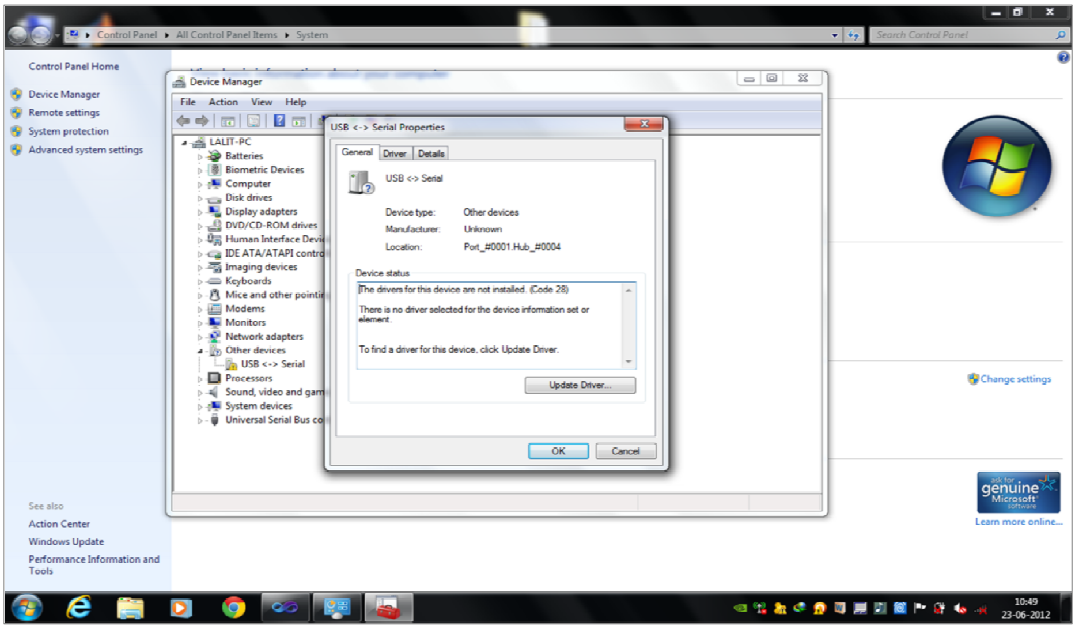


Fig. 9: USB to Serial properties

Click on the driver tab and then click on the update driver button.

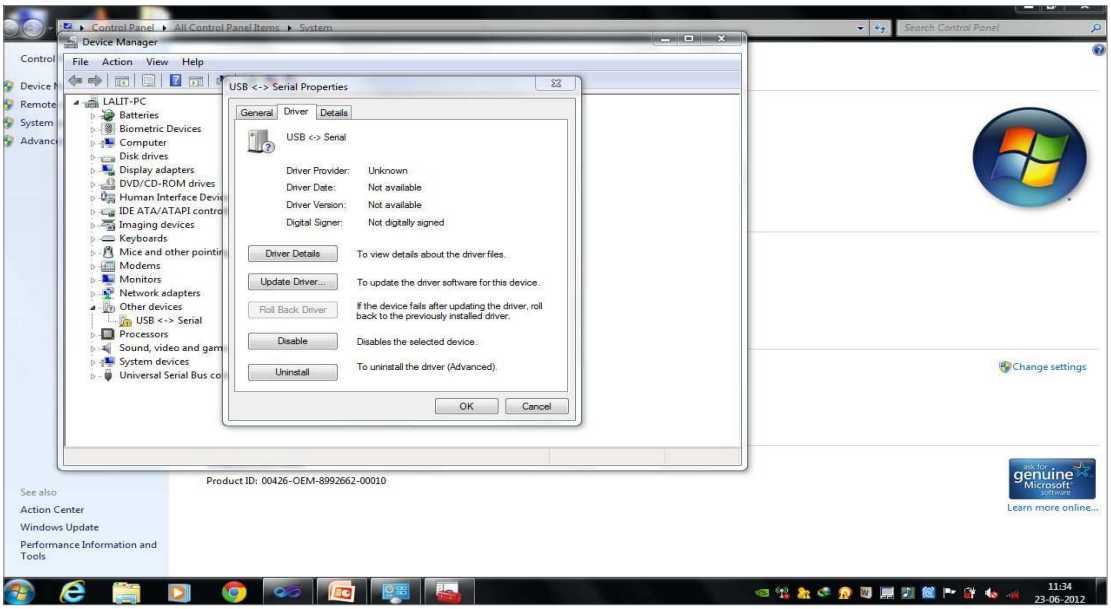


Fig. 10: Driver properties

Select 'browse my computer for driver software'

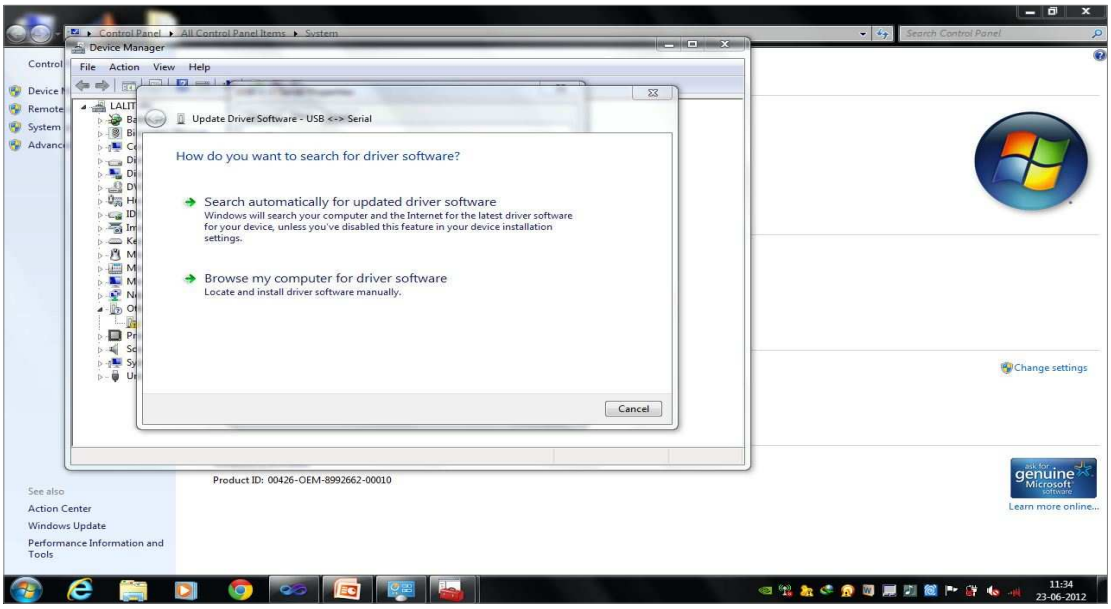


Fig. 11: Update driver

Download the driver from the CD provided along with the kit. Give the location of the driver as shown below:

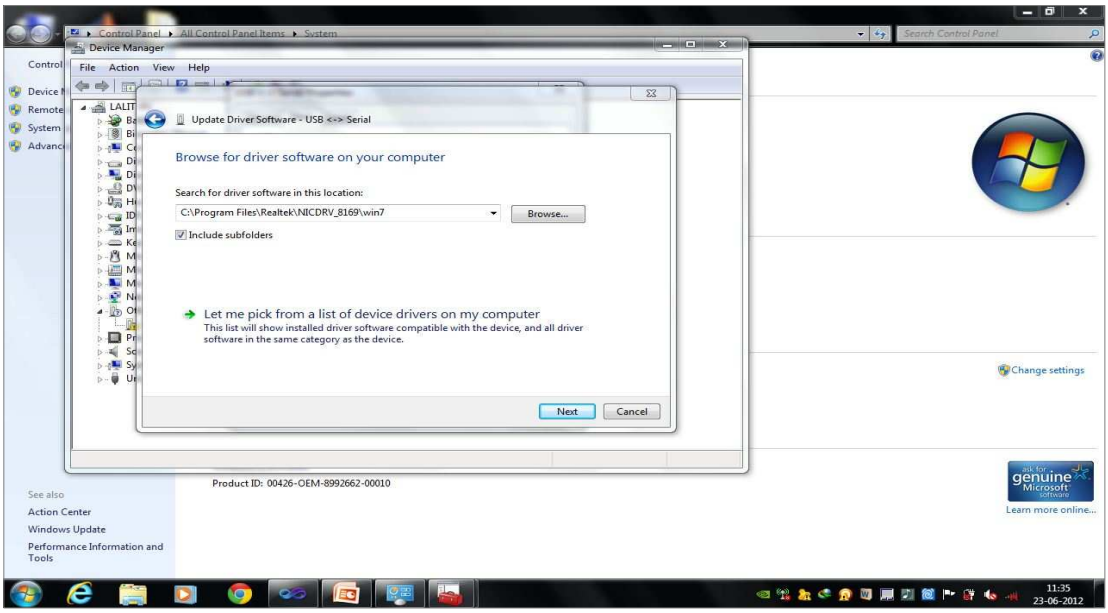


Fig. 12: Driver path

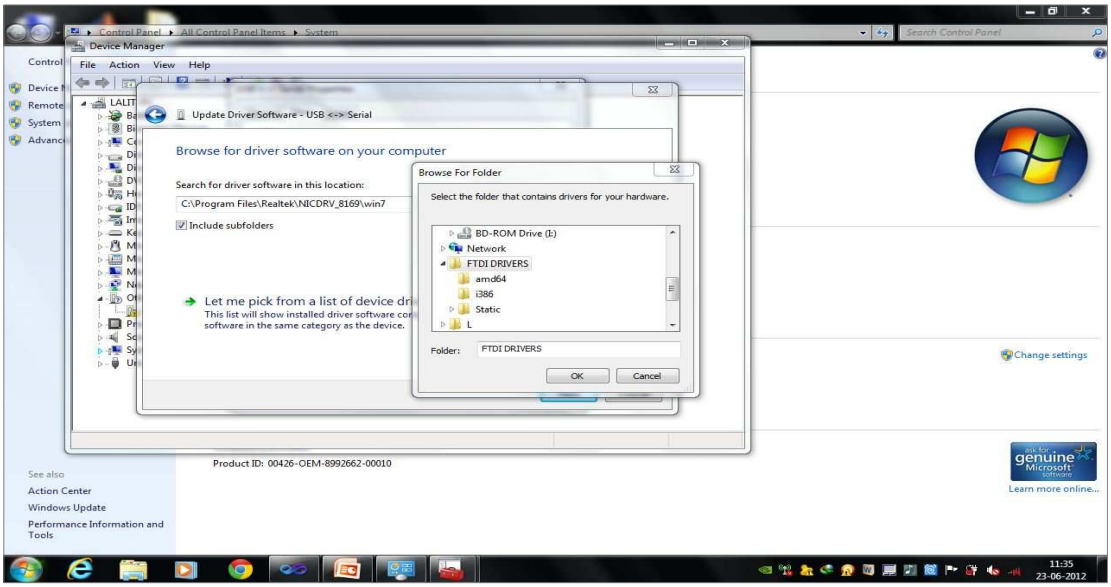


Fig. 13: Selection of driver corresponding to your computer

Installing driver software.

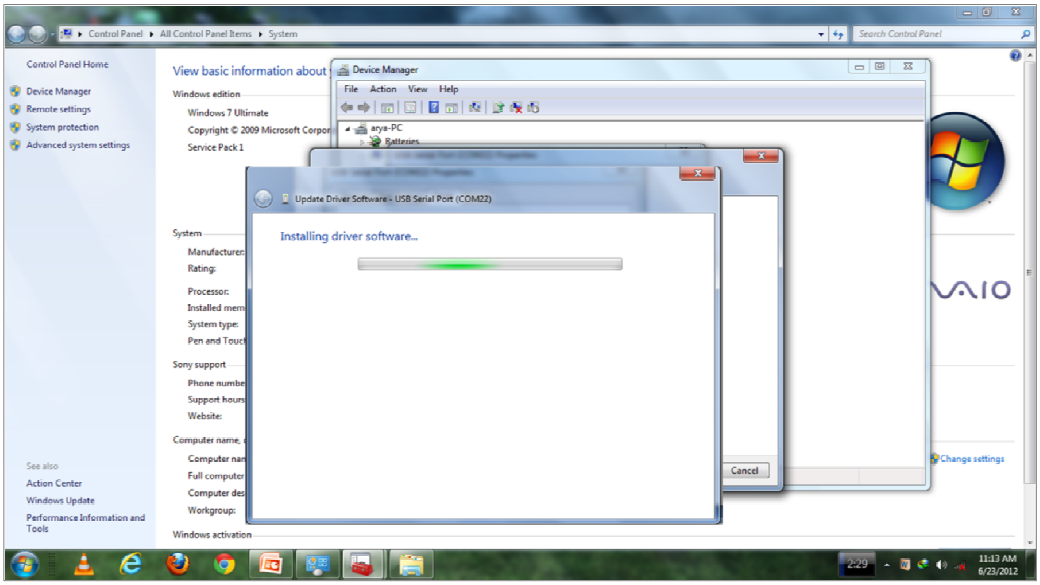


Fig. 14: Installing drivers

After the successful installation of the drivers following window will appear…

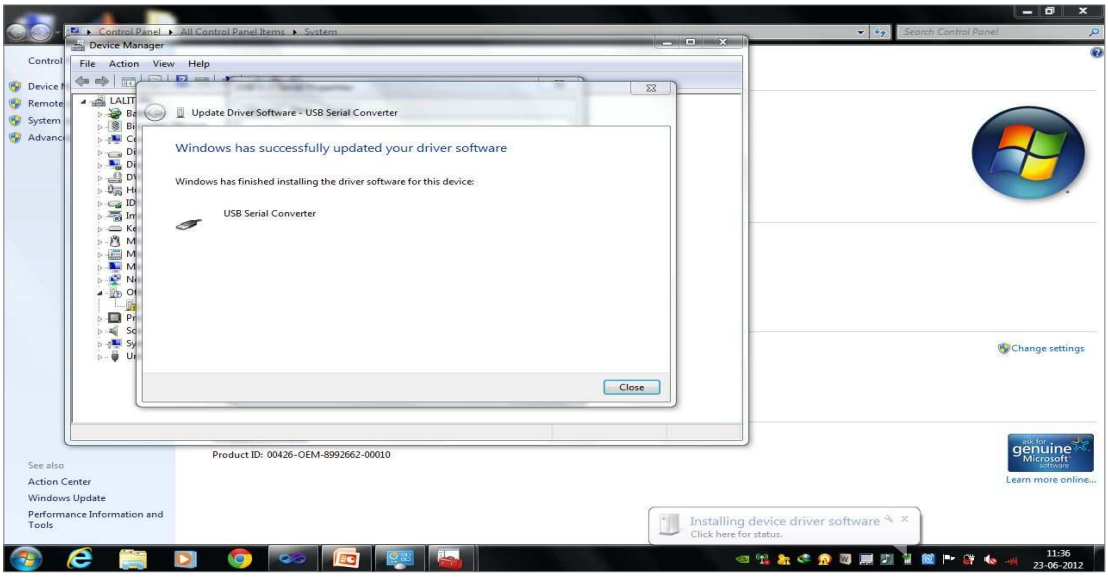


Fig. 15: Successful Installation

Close the above window, and recheck the device manager window for successful installation of the device drive software.

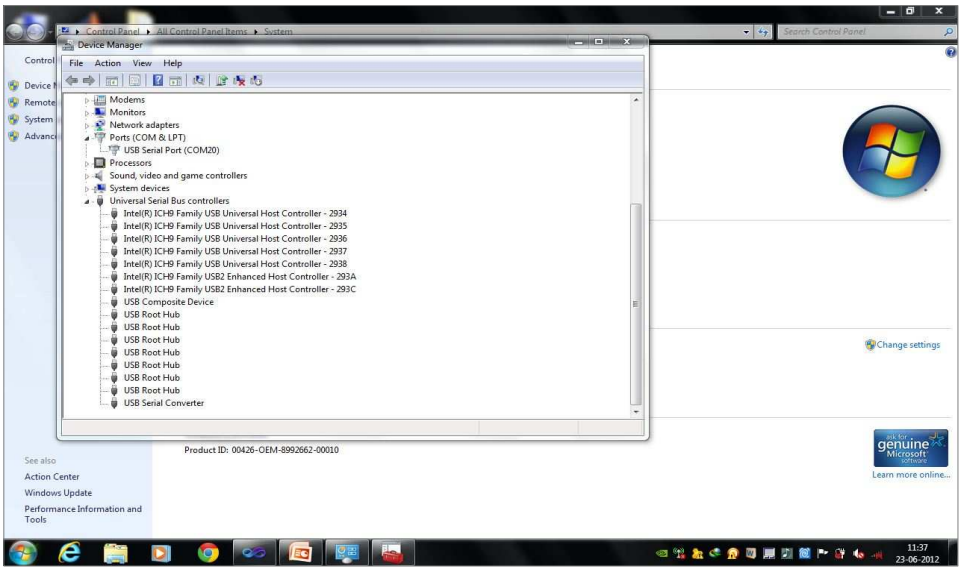


Fig. 16: Driver successfully installed

Plug out the device and connect it again and you will see the device is detected as shown below in the Fig. 17.

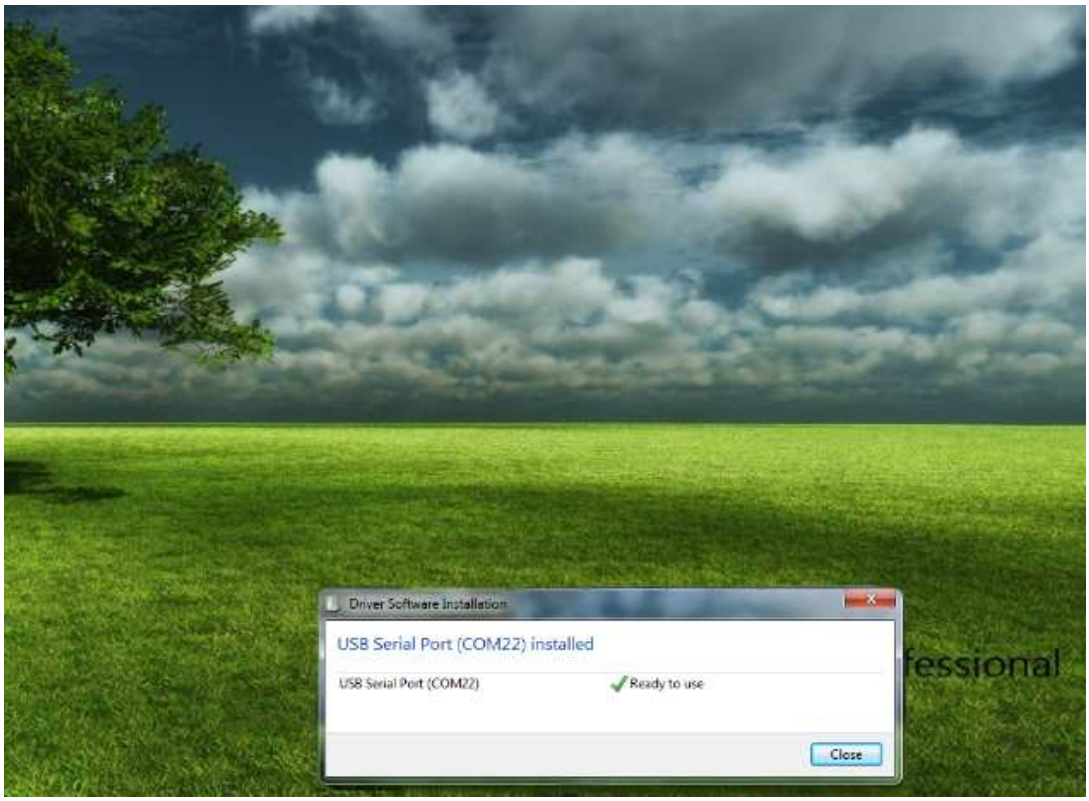


Fig. 17: Hardware detected

Programming

Basic steps:

1.Connect the USB port to your computer, and install the drivers as described above step by step.

2.Move DIP switch to its 'ON' position (programming mode).

3.Choose between XT and RC using a mini jumper as mentioned in Fig. 1 and Fig. 2.

4.You can choose between the two ways of programming (using GUI or the command prompt).

Programming with GUI interface

The pictorial description presented below explains how to program the PIC microcontroller on the board.

First install the PICPgm software.

For installing the PICPgm, click on the winpicpgm software provided in the CD and then follow the steps as shown in the figures from 18-33.

In the start the following window will appear:

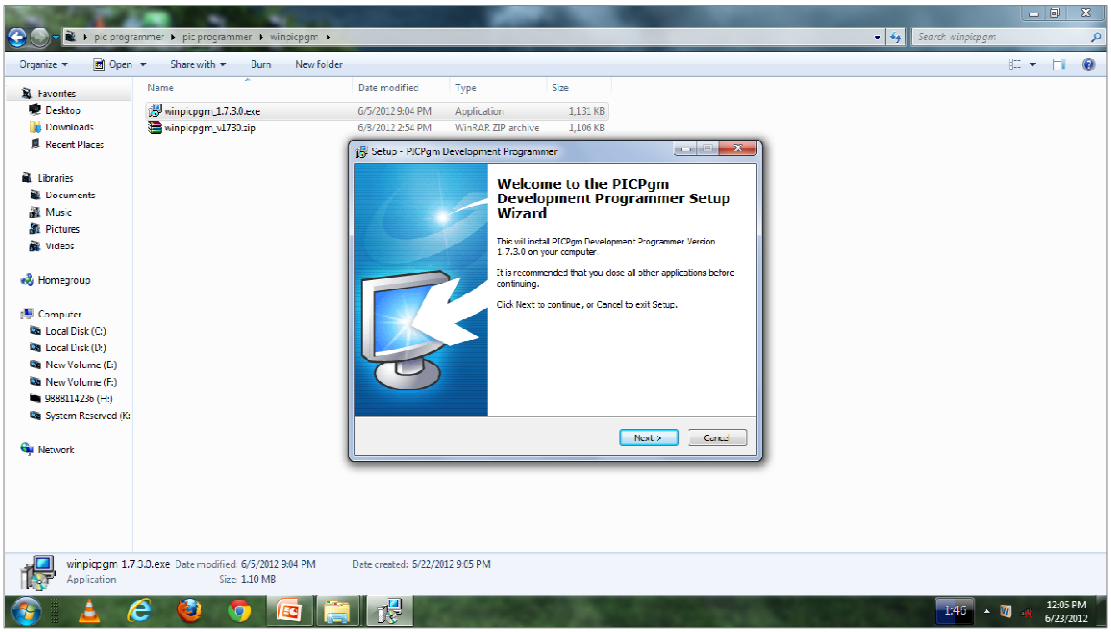


Fig. 18: Installation of PICPgm

Click **Next.**

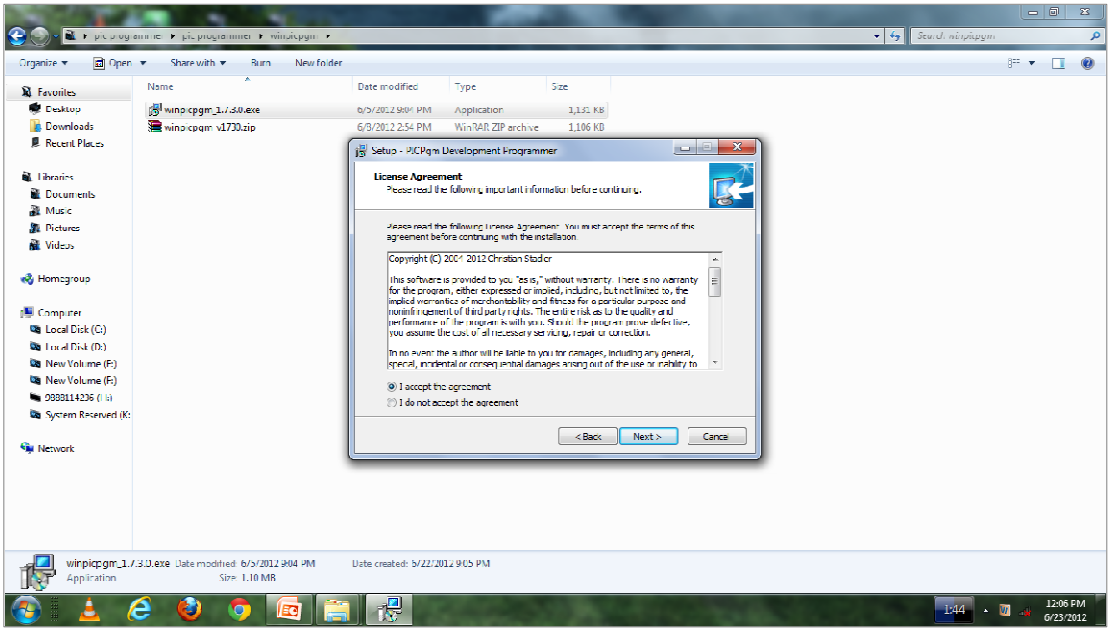


Fig. 19

Click on the 'I accept the agreement', and then click **Next.**

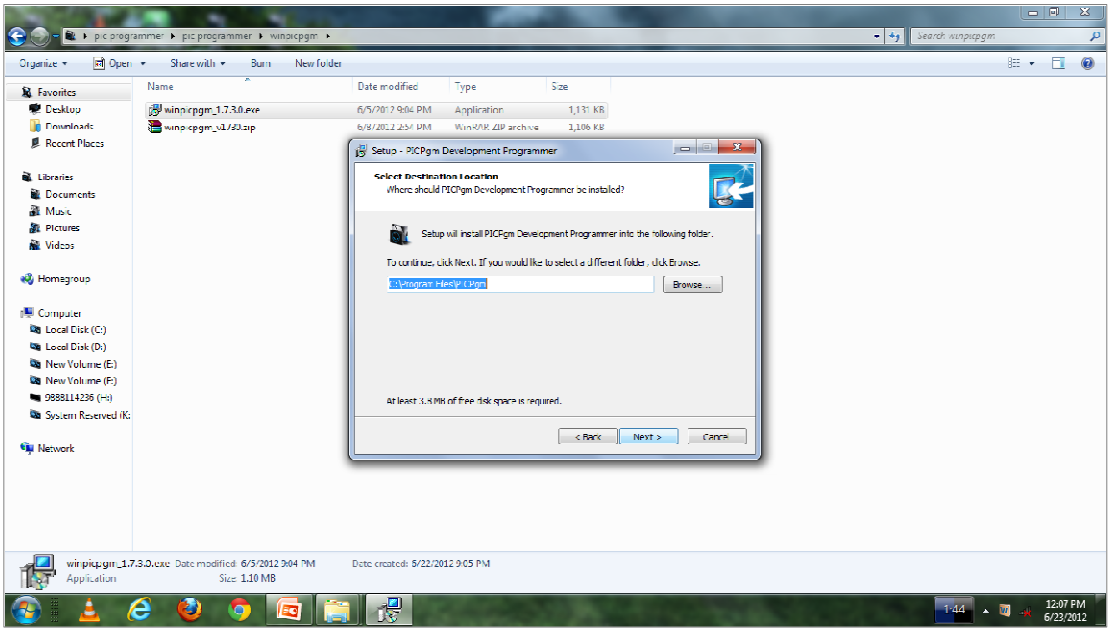


Fig. 20

Provide the destination path for the installation of the software and click **Next.**

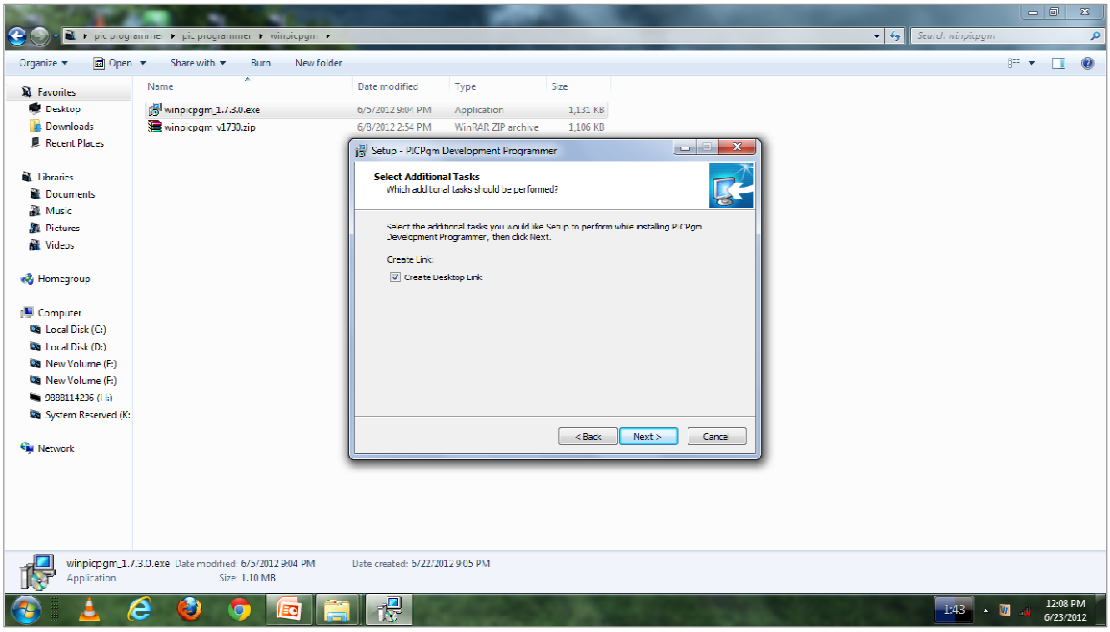


Fig. 21:

Click **Next.**

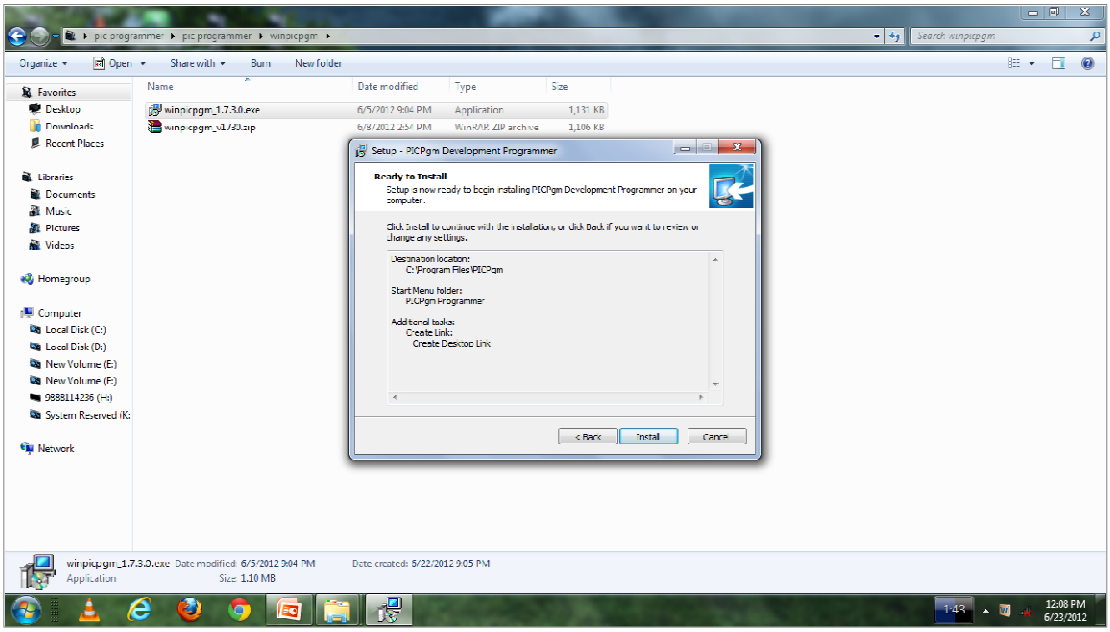


Fig. 22

Click **Install.**

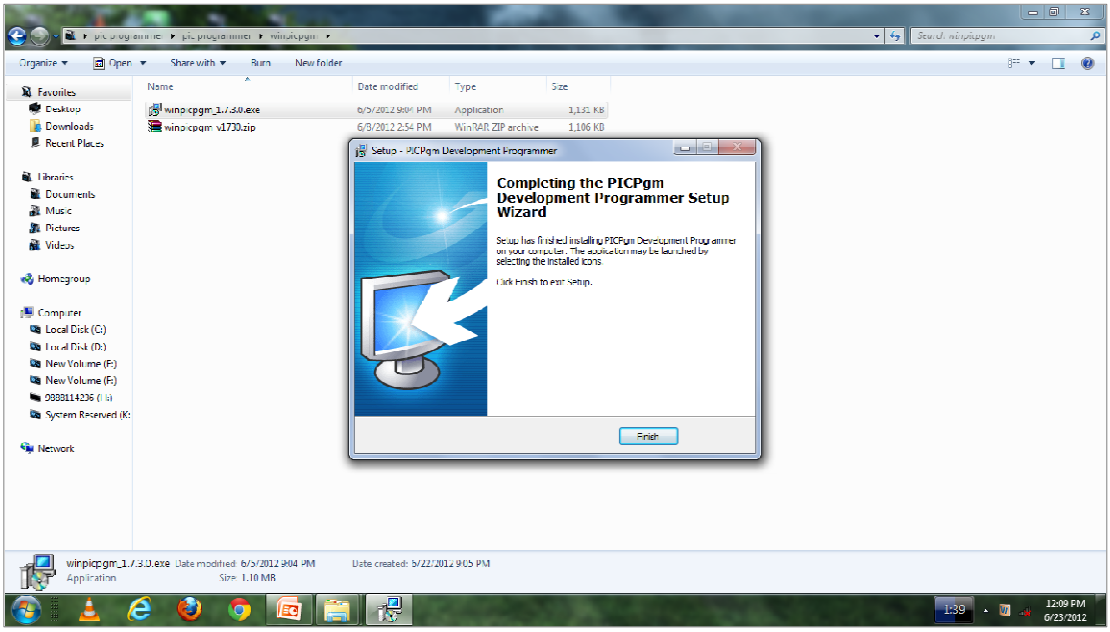


Fig. 23

Click **Finish,** to complete the installation.

After successful installation of the software, the icon of PICPgm will appear on the desktop.

Fig. 24

Click on the icon of PICPgm, if the following window appears, click **OK** and proceed for programming.

Fig. 25

After clicking **OK**, the following window of PICPgm will appear.

Fig. 26

Point to be noted here is that the above window neither indicates which PIC we are using, nor identify the programmer.

Thus click on autodetect programmer.

Fig. 27

After clicking on the programmer, the PIC microcontroller will be detected and shown on the GUI.

Fig. 28

Fig. 29

If the above widow appears somewhere in between, click **OK** and proceed to configure the bits as displayed in the Fig. 30.

Fig. 30

To program the PIC microcontroller in the XT mode, open the two pins of XT/RC header. To program in the RC mode, short both the pins using a mini jumper. Keep WDTE, PWRTE and BOREN as disabled.

Fig. 31

Browse for the hex file and click program PIC**.** The following window will show the status of programming.

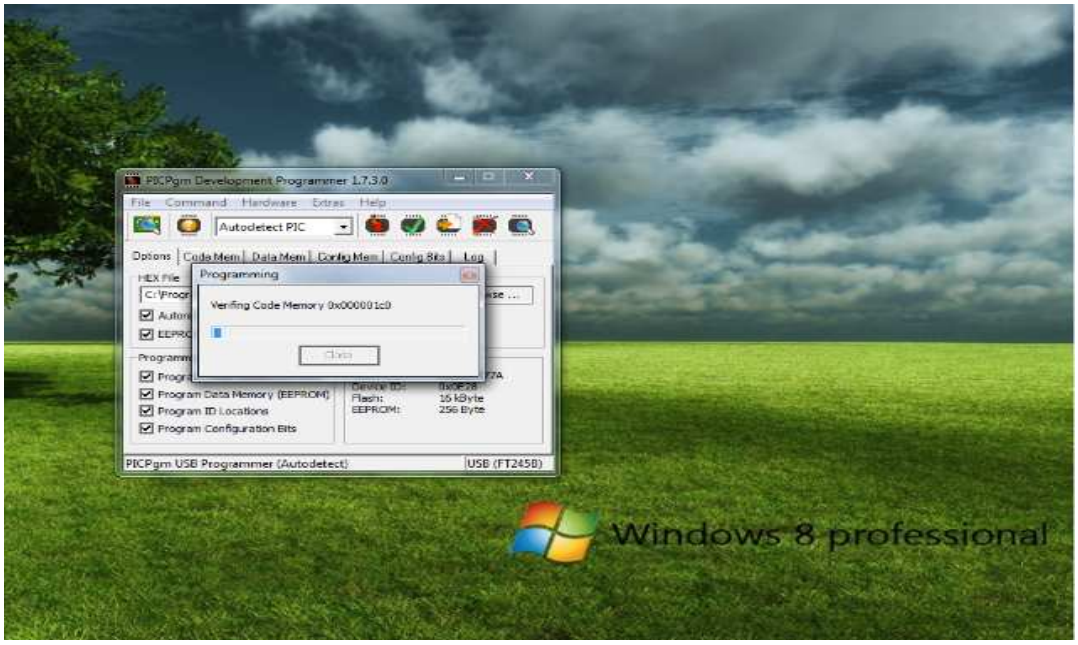


Fig. 32

After the successful completion of the programming, the following window will pop-up to indicate the program being burnt successfully.

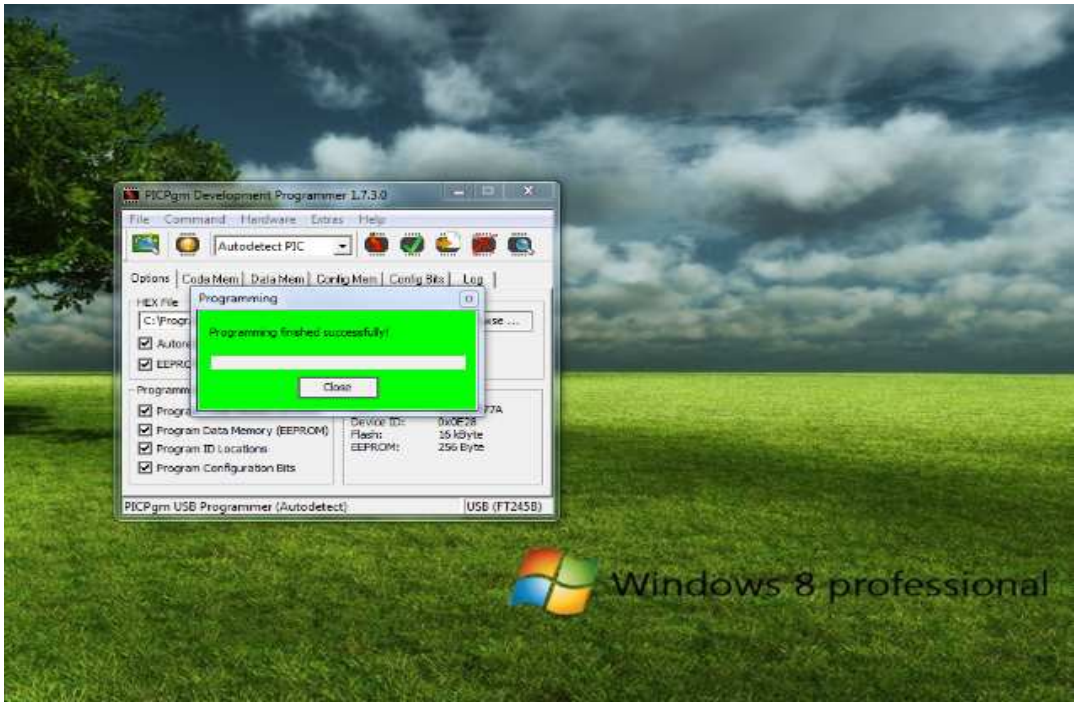


Fig. 33

Programming with command prompt

To program the PIC microcontroller using command prompt, follow the below mentioned steps:

Open the command prompt window and go to the directory having PICPgm.exe as shown below in Fig. 34.

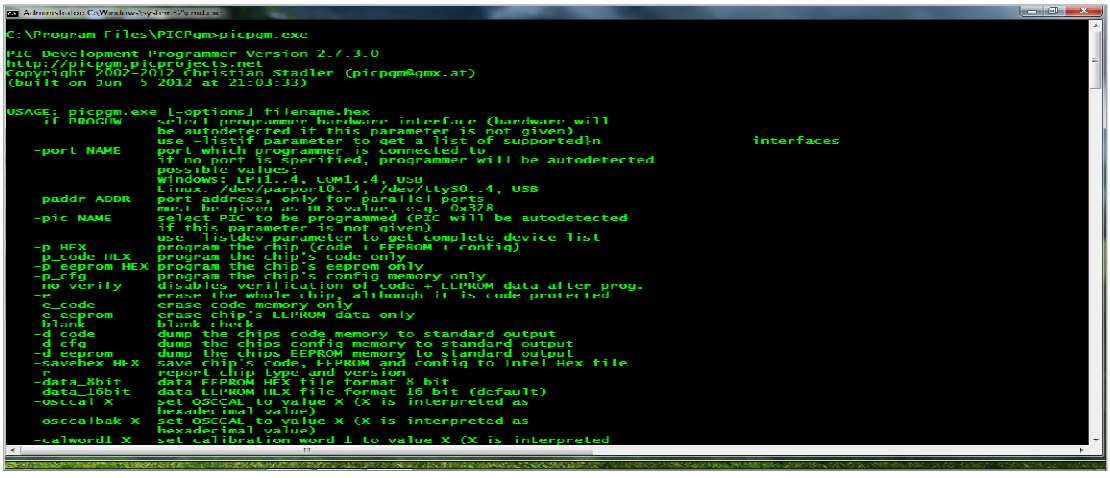


Fig. 34

To auto-detect the programmer, type '-r', after the path.

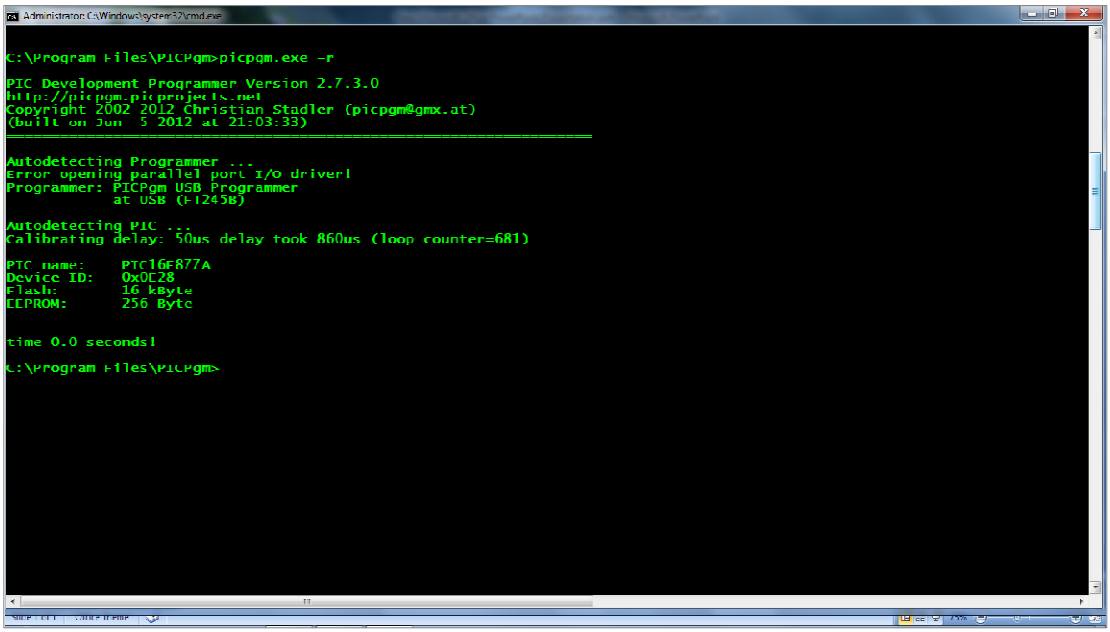


Fig. 35

To erase the previous program from the PIC controller, type '-e'.

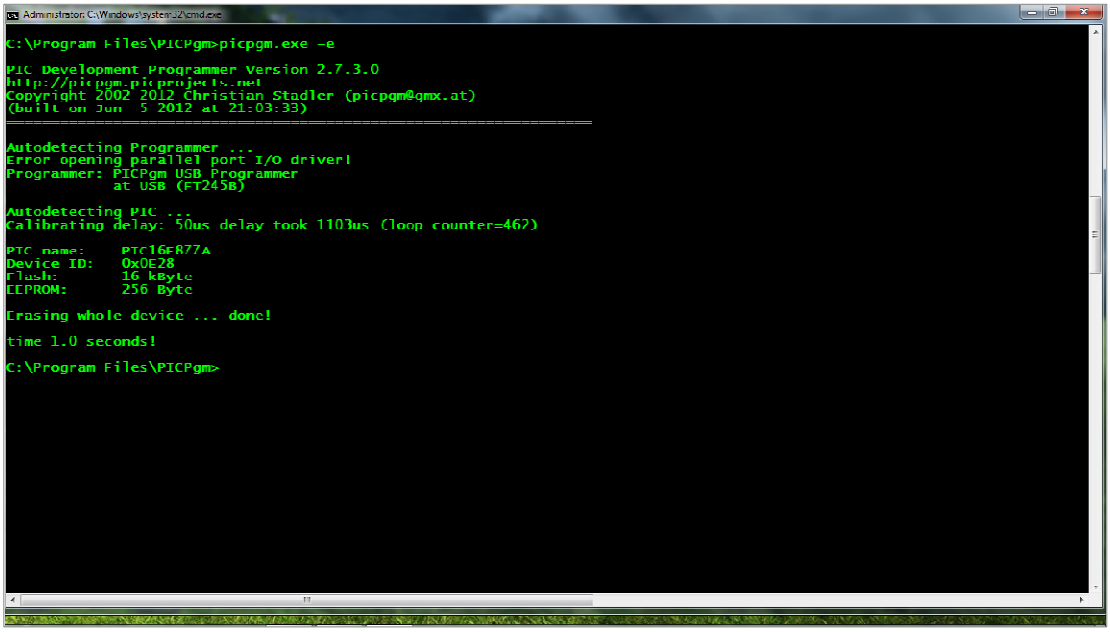


Fig. 36

To program the microcontroller, copy paste the hex file of the program into the folder where the PICpgm software is installed (C:\Program Files\PICPgm), then provide the path of the software and type '-p' and the name of the file with an extension.hex as shown below in Fig. 37.

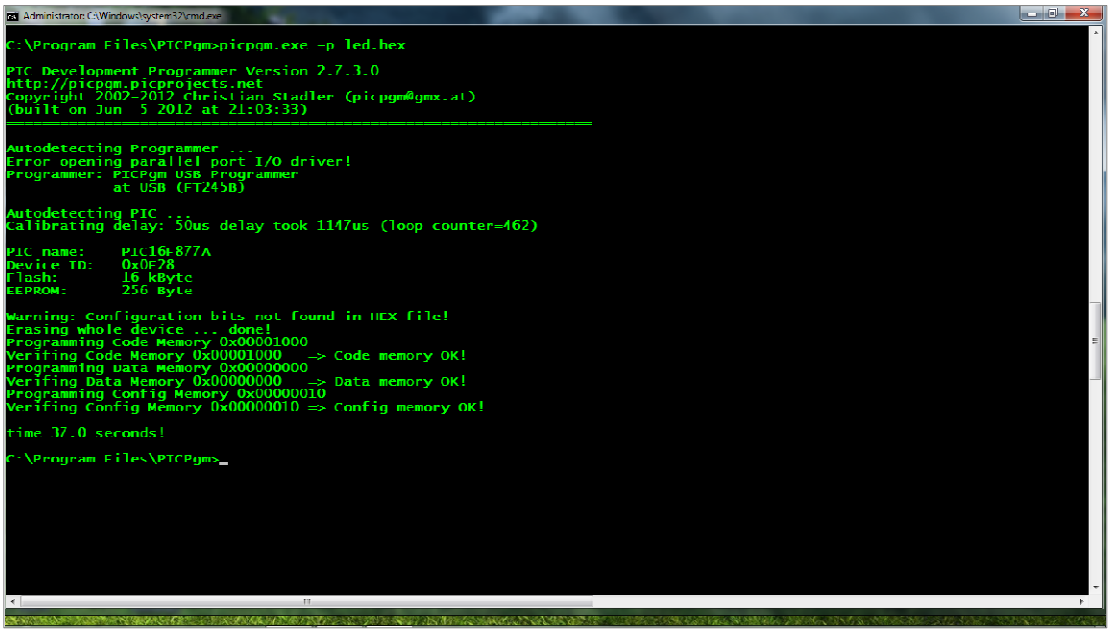


Fig. 37

Sample programs:

You can program the microcontroller with the sample code provided in the CD. The code is also copied below:

1.Connect the board to your computer using USB port.

2.Move the mode selection switch to the 'ON' position

3.Open PICPgm and follow the programming instructions to program the microcontroller

4.Move back the mode selection DIP switch to its original position and press reset.

5.The LCD and LEDs will work now. For your reference, program is also pasted below.

```
#include<pic.h>
__CONFIG(WDTDIS & XT & UNPROTECT);
char name1[32]={"  KitsNspares     KitsNspares   "};

void delay(int x)                  //Function to Provide Delay
{
   int d,l;
  for(l=0;l<x;l++)
  {
  for(d=0;d<3000;d++);
  }
}
void instwrt(void)                 //Function to write command
  {
  RC0=0;
  RC1=0;
  RC2=1;
  delay(1);
  RC2=0;
  delay(1);
}
void datawrt(void)                 //Function to write data
  {
  RC0=1;
  RC1=0;
  RC2=1;
```

```
  delay(1);
  RC2=0;
  delay(1);
  }
void lcdin()                    //Function to initialize LCD
  {
  PORTD=0x38;
  instwrt();
  PORTD=0x0c;
  instwrt();
  PORTD=0x01;
  instwrt();
  PORTD=0x06;
  instwrt();
  PORTD=0x80;
  instwrt();
  }
void main(void)
  {
  int a;
  TRISD=0x00;
  TRISC=0X00;
  TRISB=0x00;
  lcdin();
  for(a=0;a<32;a++)
  {
```

```
      PORTD=name1[a];
      datawrt();
   }
while(1)
{
   PORTD=0x18;
   instwrt();
   PORTB=0x0aa;
   delay(10);
   PORTB=0x55;
   delay(10);
}
}
```

----------END OF DOCUMENT----------