



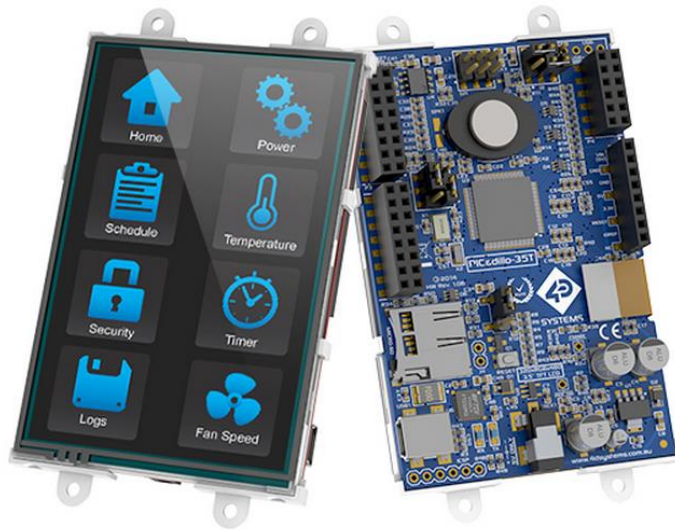
Picadillo MPLABX Initializing the Screen

DOCUMENT DATE: **22th APRIL 2019**
DOCUMENT REVISION: **1.1**



Description

The coverage of this application note demonstrates the system configuration initialization of the Picadillo-35T.



Picadillo-35T

PIC32 3.5" Display Module with Resistive Touch

- The target module is the [Picadillo-35T](#)
- [FTDI USB Chip driver](#).
- [MPLABX IDE](#) and [XC32++](#) to be downloaded and installed on a computer.
- Microchip [PICkit3 In-Circuit Debugger](#) development tool

When downloading an application note, a list of recommended application notes is shown. It is assumed that the user has read or has a working knowledge of the topics presented in these recommended application notes.

Before getting started, the following are required:

Content

Description	2
Content	3
Application Overview	4
Setup Procedure	4
Open the Project in MPLABX	5
The library project – pic35DT	5
<i>Description of content.....</i>	<i>5</i>
The Header file - pic35DT.h	5
The Source file – pic35DT.cpp	5
How the pic35T library was added to PROJECT1	6
The application project - PROJECT1.X	7
Programming the Picadillo-35T	8
<i>Setting up the PICKit3 ICD</i>	<i>8</i>
<i>Connecting the PICKit3 ICD and Picadillo-35DT</i>	<i>8</i>
Proprietary Information	10
Disclaimer of Warranties & Limitation of Liability	10

Application Overview

The Picadillo-35T is a PIC32MX795F512L based 3.5" display module with Resistive Touch. In this application note, the initial system configuration will be presented but only limited to initializing the display related peripheral and the display screen. At the end of this application note the Picadillo-35T will have an initialized screen display with a background of colour BLUE.

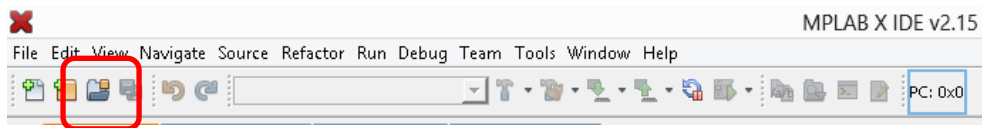
Setup Procedure

For instructions on how to install the related software development tools, compiler and device drivers please proceed to WWW.MICROCHIP.COM or click the link below for a more detailed information.

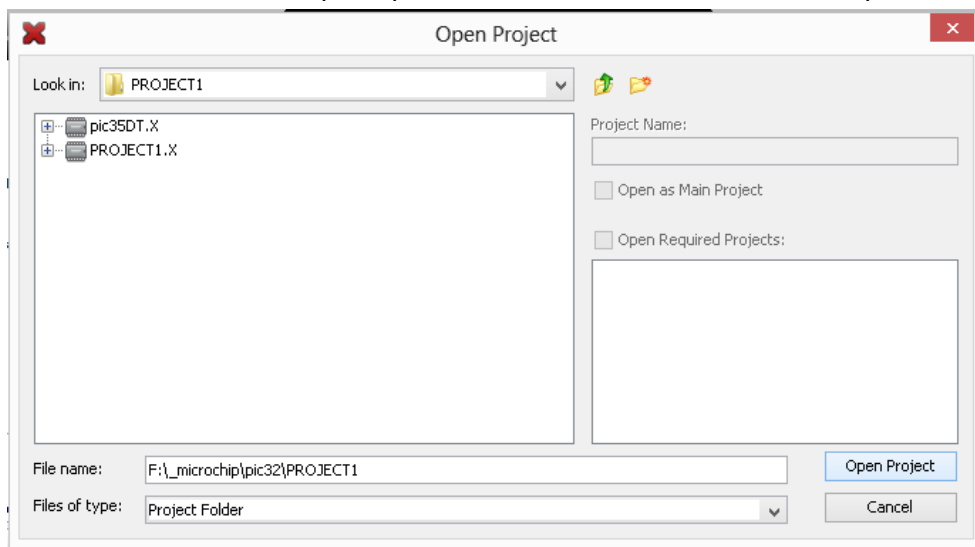
[MPLAB X Reference Materials](#)

Open the Project in MPLABX

Download the project zip file of this application note [Picadillo MPLABX Initializing the Screen](#). After completing the download extract the files to the preferred location.



Afterwards, the IDE will prompt the user to locate the files to be opened.



Select the target project file named – PROJECT1.X. Also, open the project library file – pic35DT. Further details about the project files will be presented in the succeeding parts of this application note.

The library project – pic35DT

Description of content

The Header file - pic35DT.h

The header file in this library project is uses the C++ programming language. This necessitates the use of the XC32++ for pic32 microcontrollers. This header files has all the prototype functions needed to initialize the Picadillo-35T screen display. It also contains several other commands that will simplify drawing of primitive shapes that are to be displayed on the TFT screen.

This header file also includes another header file that contains the RGB-565 colours that are precisely toned to bring about the colour vividly. Looking inside the library project the user will find the colourPalette.h file. In here the list of colour definitions.

The Source file – pic35DT.cpp

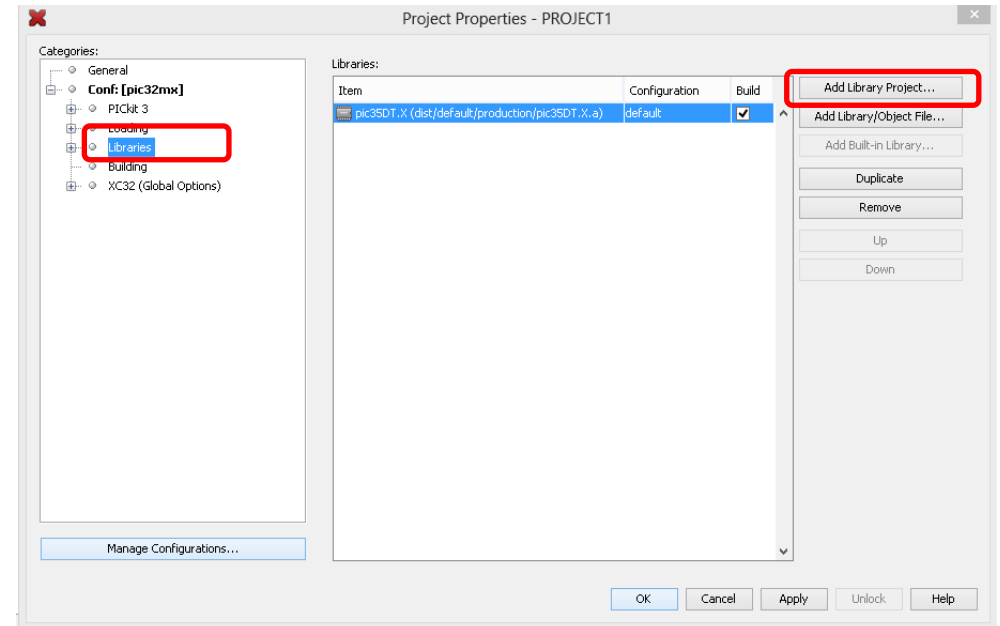
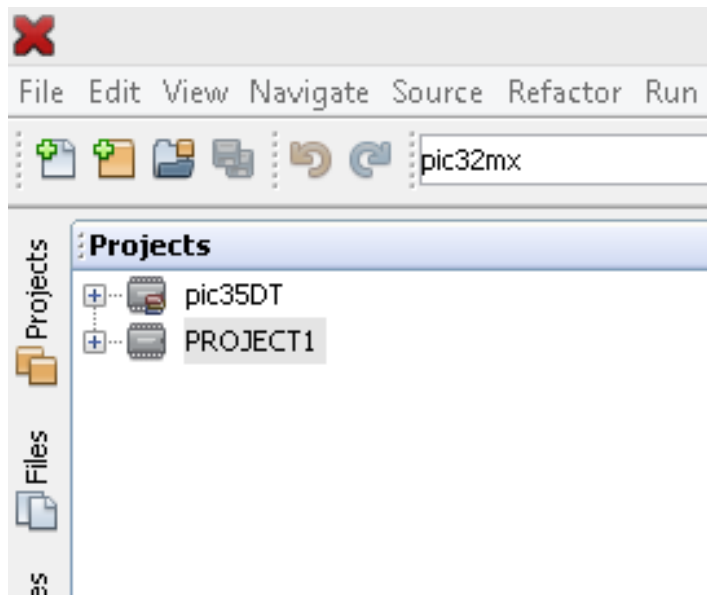
The command prototypes listed in the header files have all their declaration inside this file. This is important and must always be together with the header files mentioned above.

This file contains the most important function for this project, the `init_lcd()`. This function initially setup the PMP port of the PIC32MX795F512L and also the GPIO that controls the BACKLIGHT. For this library project, no PWM is applied to the backlight GPIO to simplify the process of learning.

How the pic35T library was added to PROJECT1

This section explains how the project was added to the primary project file – PROJECT1.X. On the MPLABX projects window, right click on the folder PROJECT1. A pop-up window will appear and then select the PROPERTIES.

Selecting the PROJECT PROPERTIES option on the pop-up menu will open another window. This window contains settings that can be categorized into two parts- the main configuration setup and the General setup options.



Under the libraries options, the project was added through the 'ADD LIBRARY PROJECT...' option. The folder was then located and added.

****FOR THIS APPLICATION NOTE THIS SECTION WILL HAVE TO CHECK TO CORRECTLY LOCATE THE DIRECTORY OF THE PIC35DT LIBRARY PROJECT. NOT DOING SO WILL RESULTS TO COMPILATION ERRORS.**

The application project - PROJECT1.X

This application starts with the setup of device configuration. The device configuration can be varied according to the line of application intended for the display. The device configuration is shown below.

```
#pragma config FSRSEL = PRIORITY_7
#pragma config FMIIEN = OFF
#pragma config FETHIO = OFF
#pragma config FCANIO = OFF
#pragma config FUSBIDIO = OFF
#pragma config FVBUSONIO = OFF
```

```
#pragma config FPLLIDIV = DIV_3
#pragma config FPLLMUL = MUL_20
#pragma config UPLLIDIV = DIV_4
#pragma config UPLEN = OFF
#pragma config FPLLODIV = DIV_1
```

```
#pragma config FNOSC = PRIPLL
#pragma config FSOSCEN = OFF
#pragma config IESO = OFF
#pragma config POSCMOD = XT
#pragma config OSCIOFNC = OFF
#pragma config FPBDIV = DIV_8
#pragma config FCKSM = CSECMD
#pragma config WDTPS = PS1048576
#pragma config FWDTEN = OFF
```

```
// DEVCFG0
#pragma config DEBUG = OFF
#pragma config ICESEL = ICS_PGx2
#pragma config PWP = OFF
#pragma config BWP = OFF
#pragma config CP = OFF
```

Following the device configuration, pre-processors were added to the application. Among these is the pic35DT.h file

```
#include <cstdlib>
#include <xc.h>
#include "../pic35DT.X/pic35DT.h" //locate directory of header
```

**** For this section the user will have to relocate the directory location of the file. NOT DOING SO WILL CAUSE ERROR DURING BUILDING/COMPILING.**

The main body of PROJECT1, is composed of simple commands that are enlisted in the pic35DT library project. The init_lcd() function contains the setup for the TFT initialization commands. These commands in the form of HEX values are passed to the screen over at the PMP of the device controller more information on this peripheral please refer the PIC32MX Family reference manual.

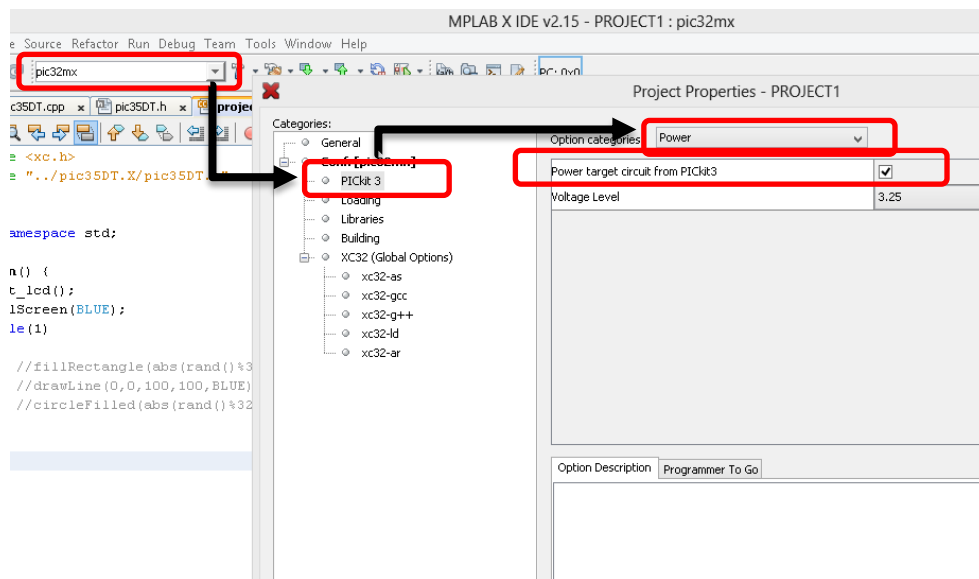
```
int main() {
    init_lcd();
    fillScreen(BLUE);
    while(1)
    {
        //fillRectangle(abs(rand()%320),abs(rand()%480),abs(rand()%320),abs(rand()%480),abs(rand()%0xFFFF));
        //drawLine(0,0,100,100,BLUE);
        //circleFilled(abs(rand()%320),abs(rand()%480),abs(rand()%100),abs(rand()%0xFFFF));
    }
}
```

If desired the user can uncomment the function in the while() loop to see the resulting display on the screen.

Programming the Picadillo-35T

Setting up the PICKit3 ICD

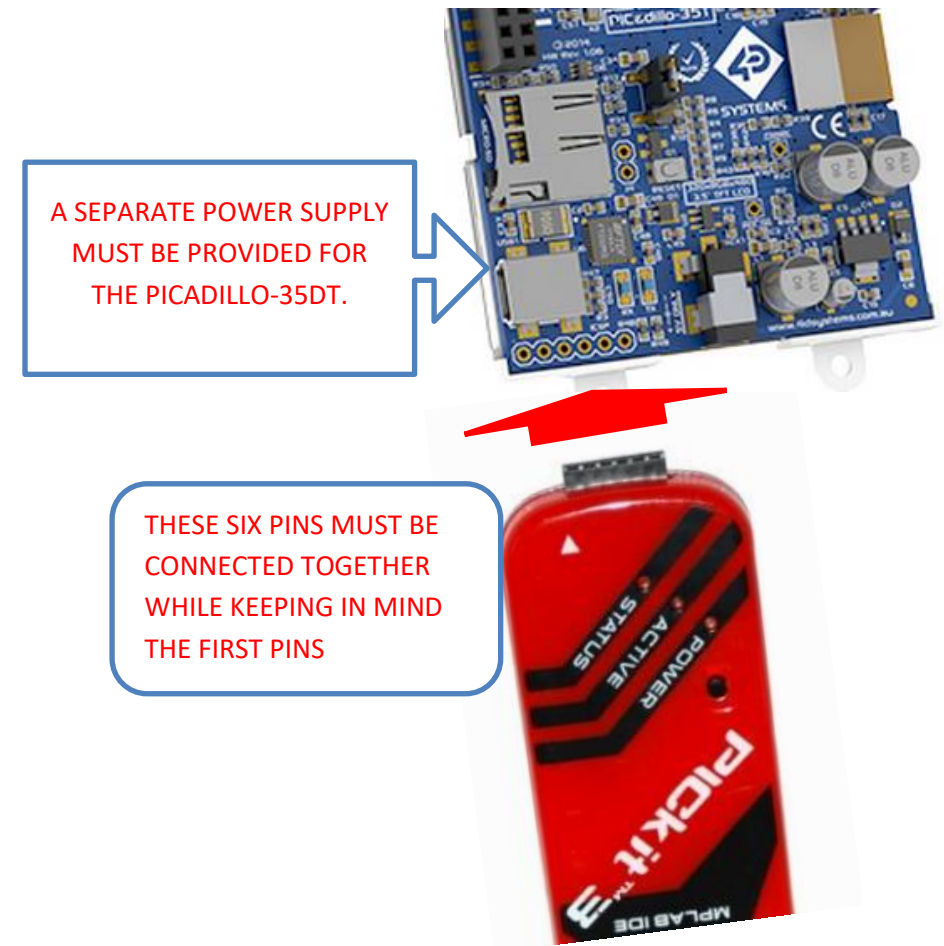
After successfully re-building the project, the project is now ready to be downloaded to the display module using a PICKit3 In-Circuit Debugger. First, we need to make sure that our PICKit3 is setup properly. The PICKit power should be enabled to power the target circuit. This can be done by following the steps below.



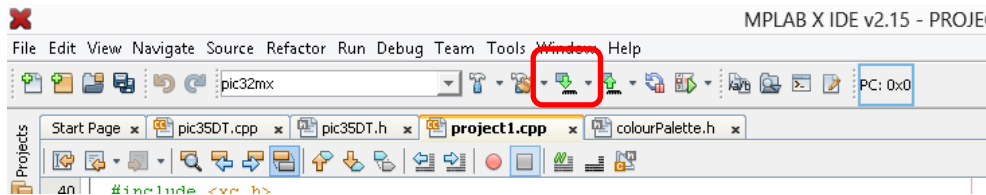
Having done the aforementioned we can now proceed to connecting the PICKit3 to the Picadillo-35DT

Connecting the PICKit3 ICD and Picadillo-35DT

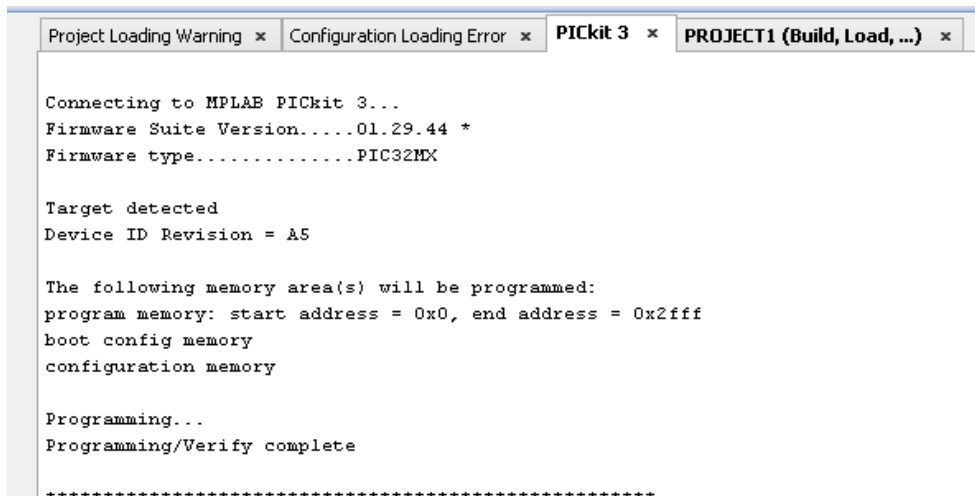
Follow the image below as a reference for the connection between the two devices.



After all the setup and connections are made, the last thing to do is to download the project.



Click the icon in the red box to start the program download. MPLAB will first make sure that everything in the code is correct by doing a compile process and then proceeds to downloading.



If the image above is shown in the MPLABX IDE then this means that everything finished without any problems.

Proprietary Information

The information contained in this document is the property of 4D Systems Pty. Ltd. and may be the subject of patents pending or granted, and must not be copied or disclosed without prior written permission.

4D Systems endeavours to ensure that the information in this document is correct and fairly stated but does not accept liability for any error or omission. The development of 4D Systems products and services is continuous and published information may not be up to date. It is important to check the current position with 4D Systems.

All trademarks belong to their respective owners and are recognised and acknowledged.

Disclaimer of Warranties & Limitation of Liability

4D Systems makes no warranty, either expresses or implied with respect to any product, and specifically disclaims all other warranties, including, without limitation, warranties for merchantability, non-infringement and fitness for any particular purpose.

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications.

In no event shall 4D Systems be liable to the buyer or to any third party for any indirect, incidental, special, consequential, punitive or exemplary damages (including without limitation lost profits, lost savings, or loss of business opportunity) arising out of or relating to any product or service provided or to be provided by 4D Systems, or the use or inability to use the same, even if 4D Systems has been advised of the possibility of such damages.

4D Systems products are not fault tolerant nor designed, manufactured or intended for use or resale as on line control equipment in hazardous environments requiring fail – safe performance, such as in the operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life support machines or weapons systems in which the failure of the product could lead directly to death, personal injury or severe physical or environmental damage ('High Risk Activities'). 4D Systems and its suppliers specifically disclaim any expressed or implied warranty of fitness for High Risk Activities.

Use of 4D Systems' products and devices in 'High Risk Activities' and in any other application is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless 4D Systems from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any 4D Systems intellectual property rights.