

Pico Display Layout Builder (PDLB)

User Guide

- Desktop GUI software program to generate advanced graphics for PicoMite running on a Raspberry Pi Pico board.
- Companion software for the Pico Display Base Board.
- The software is intended to help the designer “avoid the blank-page...where do I start syndrome”.

Version 1.0

Contents

Product Overview.....1

Pico Display Layout Builder (PDLB) Description.....3

PDLB Tutorial.....6

PDLB Main Menu Reference.....8

Miscellaneous.....11

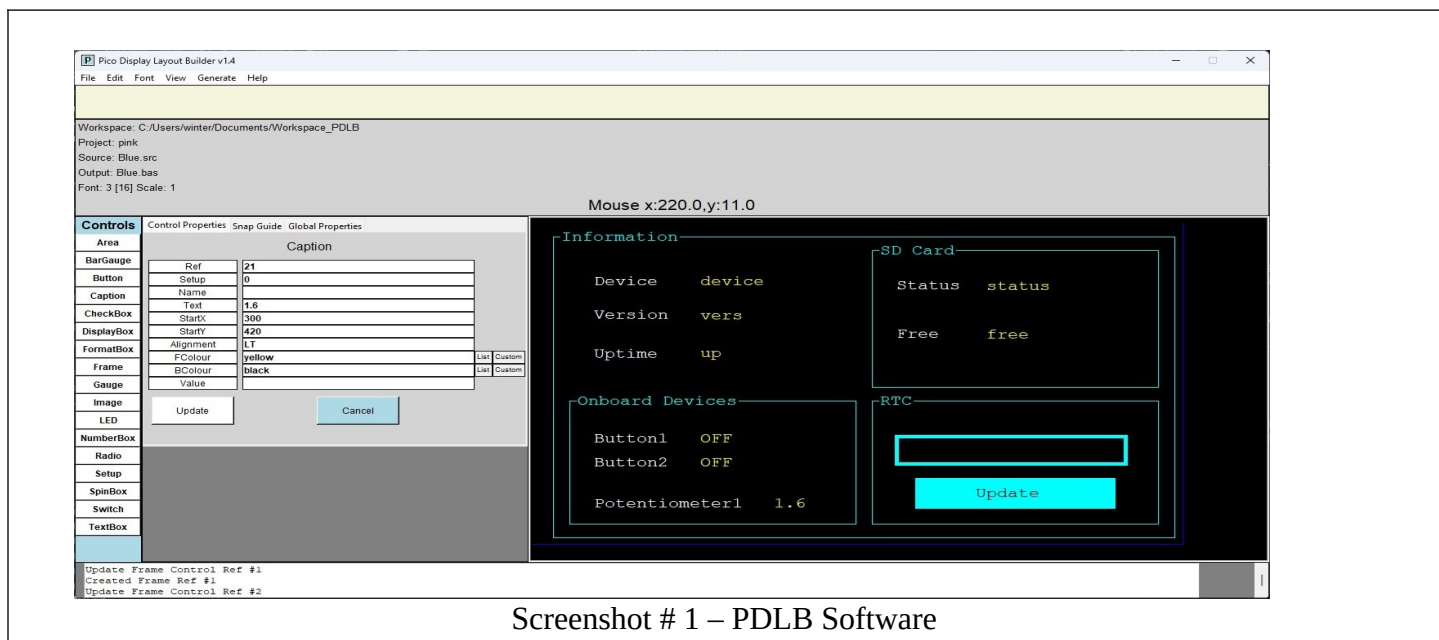
See endnotes for references to all products and authors.

Product Overview

The Pico Display Layout Editor (PDLB) software helps the hardware enthusiast design colorful displays for their Project.

Pico Display Layout Builder (PDLB) is a desktop Graphical User Interface (GUI) application to layout PicoMite advanced graphics Controls. The software ‘mimics’ the PicoMite Controls using rectangle and circle shapes which can be positioned or resized in a virtual display. When satisfied, a mouse button click generates the PicoMite BASIC code.

The product is companion software for the Pico Display Base Board (available at Tindie.) The Pico, wires and external devices are not included. However these low-cost components are available from popular online retailers.



FEATURES

- GUI Layout Builder of PicoMite advanced graphic Controls.
- Organize PicoMite Controls in files for single or multiple page displays. Merge with optional code files.
- Manage graphical design work as source text files within projects. (No database required.)
- Generates MMBasic text files that can be deployed into a PicoMite using standard utilities.

APPLICATIONS

- Supports PicoMite on the Pico Display Base Board.
 - Raspberry Pi Pico
 - PicoMite 5.07
 - SSD1963 5-inch LCD Display (800*480)
 - Integrated Touch control and SD card
- Compatible with TeraTerm and MM Edit/MM Console.

Specifications

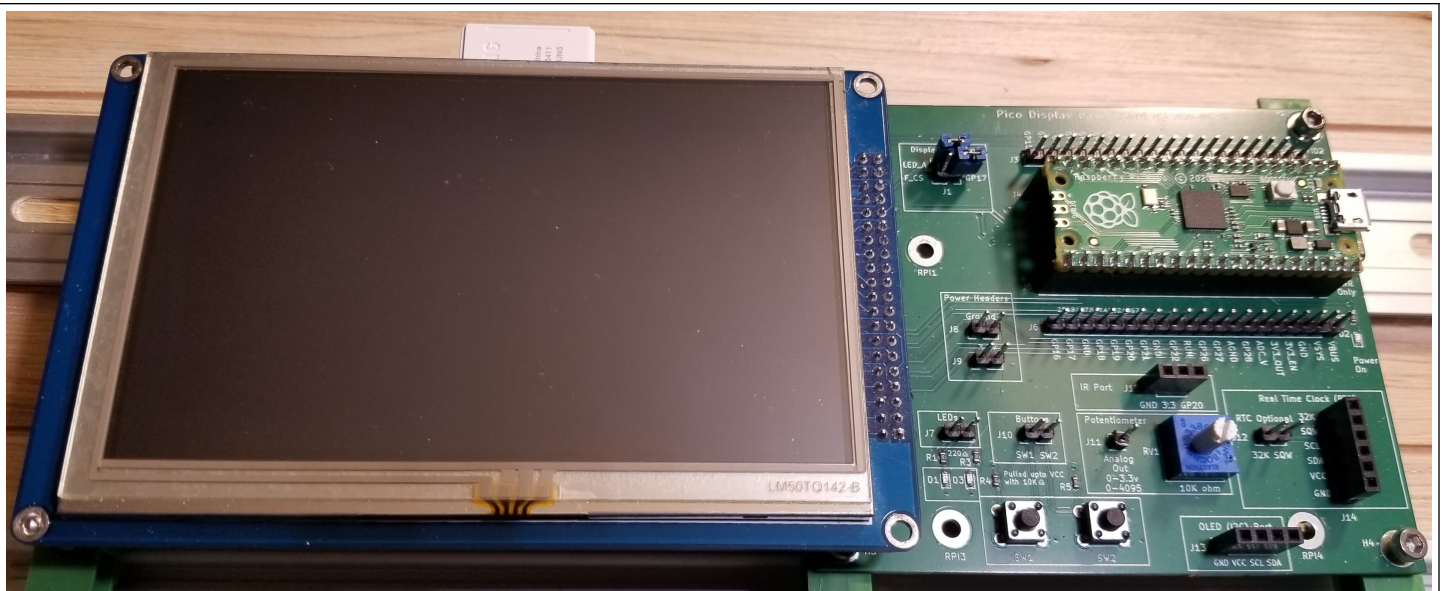
Prerequisites

The PDLB platform (Picture #1) expects the following:

- Pico Display Base Board with a Raspberry Pi Pico and the PicoMite 5.07 firmware.
- 5-inch SSD1963 graphics card plugged into the 40-pin display port.
- The SSD1963 display is integrated with a Touch screen and SD Card.
-

The kit is built on Microsoft Windows 11. Briefly tested on Windows 10 Home; found it runs just as well.

Platform:



Picture #1 – Raspberry Pi Pico on Pico Display Base Board with 5-inch SSD1963 LCD Display

Limitations

The software allows a designer to place similarly-shaped graphic rectangles and ovals (See Screenshot #3) on a virtual desktop canvas to represent PicoMite Controls. The virtually displayed objects are a close approximation to PicoMite Controls.

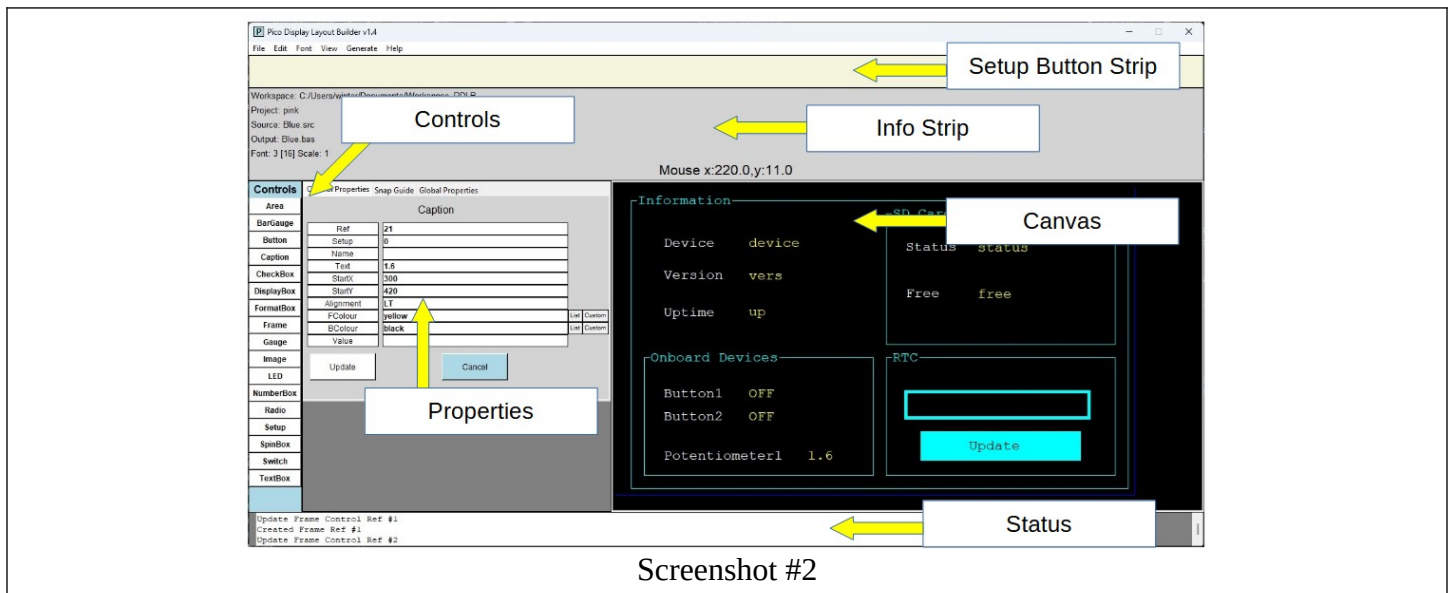
Some current limitations:

- Object selection and movement is limited to one Control at a time. (No lasso.)
- The PicoMite LCD Display is set in Landscape mode.
- The BLIT command, Sounds and Custom Fonts are not directly supported.

Pico Display Layout Builder (PDLB) Description

The PDLB software is a desktop GUI Integrated Development Environment (IDE) application that resides as a single screen with tabs and menus that manage all of the steps to design graphical layouts and generate the PicoMite code.

Below is a screenshot of the PDLB IDE with an overlay describing the various sections:



The IDE

The sections of the IDE (see Screenshot #2) are organized so the designer can concentrate on designing. Clicking one of the **Controls** (buttons) initiates a placement session with one of the PicoMite Advance Graphic Controls. The placement occurs on the **Canvas** which is a (800*480 pixel) virtual display (that is 1:1 to the PicoMite display.) Each Control is specified via a **Control Properties** pane; modifications are immediately reflected on the Canvas. The Properties pane includes dialog displays that allow the user to choose one of the PicoMite's standard colors or use the custom RGB color picker. The **Info Strip** prominently displays the mouse cursor's "X,Y" locations and the current file management values for the workspace, project, and source file. A unique feature of PicoMite is grouping Control objects on a display page into a Setup "group"; each Setup page can be viewed/edited by clicking a button in the **Setup Button Strip**. Finally, the **Status** bar displays important messages as the layout progresses.

Layout on the Canvas

Graphical objects, that mimic PicoMite Controls are depicted in the Canvas in a 1:1 relationship to those on the LCD display. Therefore, the designer can grab an object and move it around the screen to decide on its best location, re-size it, and change its color. As the designer places an object, the software maintains a textual

specification of each object, which can be saved into a text source file. Likewise, when ready, a mouse click causes the software to generate the display into PicoMite MMBasic text commands.

Canvas



For example, in the close-up of the canvas in Screenshot #3, the cyan-filled rectangle represents the layout of a Button Control (with the text and shape that it would appear on the PicoMite display.) The empty cyan-outlined rectangle represents the layout of a DisplayBox Control (which begins empty on the PicoMite but is updated with the current date/time during program execution.) The thin cyan-colored rectangles represent the Frame Control (with its caption).

Generated PicoMite Controls

At any time, the designer can click the “Build Controls” menu item, and a text file is created with the PicoMite Control MMBASIC commands (see Code #1). An alternate menu item will merge a user created code file.

```
'Pico Display Base Board
'Control layout configured: 2024-09-08 12:19:38.
'
GUI DELETE ALL
cls RGB(black)
COLOUR RGB(white), RGB(black)
FONT 3,1
```

GUI SETUP 1

GUI CAPTION #1,"Page 1",700,20,"LT",RGB(white),RGB(black)

GUI FRAME #2,"Information",20,60,770,400,RGB(pink)

GUI FRAME #3,"Frame",350,90,200,200,RGB(pink)

GUI BUTTON #4,"Next Page",570,390,200,50,RGB(black),RGB(pink)

GUI CAPTION #9,"Device",80,120,"LT",RGB(white),RGB(black)

GUI CAPTION #10,"Version",80,180,"LT",RGB(white),RGB(black)

GUI CAPTION #11,"Uptime",80,240,"LT",RGB(white),RGB(black)

GUI SETUP 2

GUI CAPTION #5,"Page 2",700,20,"LT",RGB(white),RGB(black)

GUI FRAME #6,"Configure",30,60,750,400,RGB(cyan)

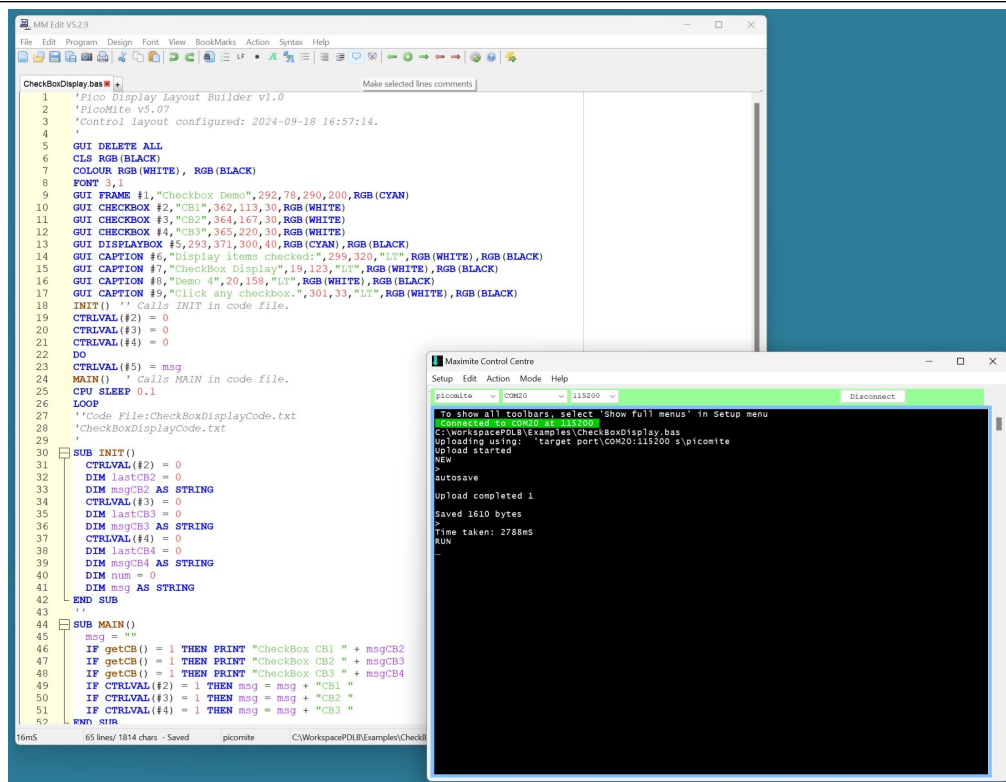
GUI BUTTON #7,"Home Page",40,390,200,50,RGB(black),RGB(cyan)

GUI CAPTION #8,"num",380,240,"LT",RGB(white),RGB(black)

Code #1

The code BAS file can be generated as “Controls Only” or with “Controls with Code” (which includes defaults).

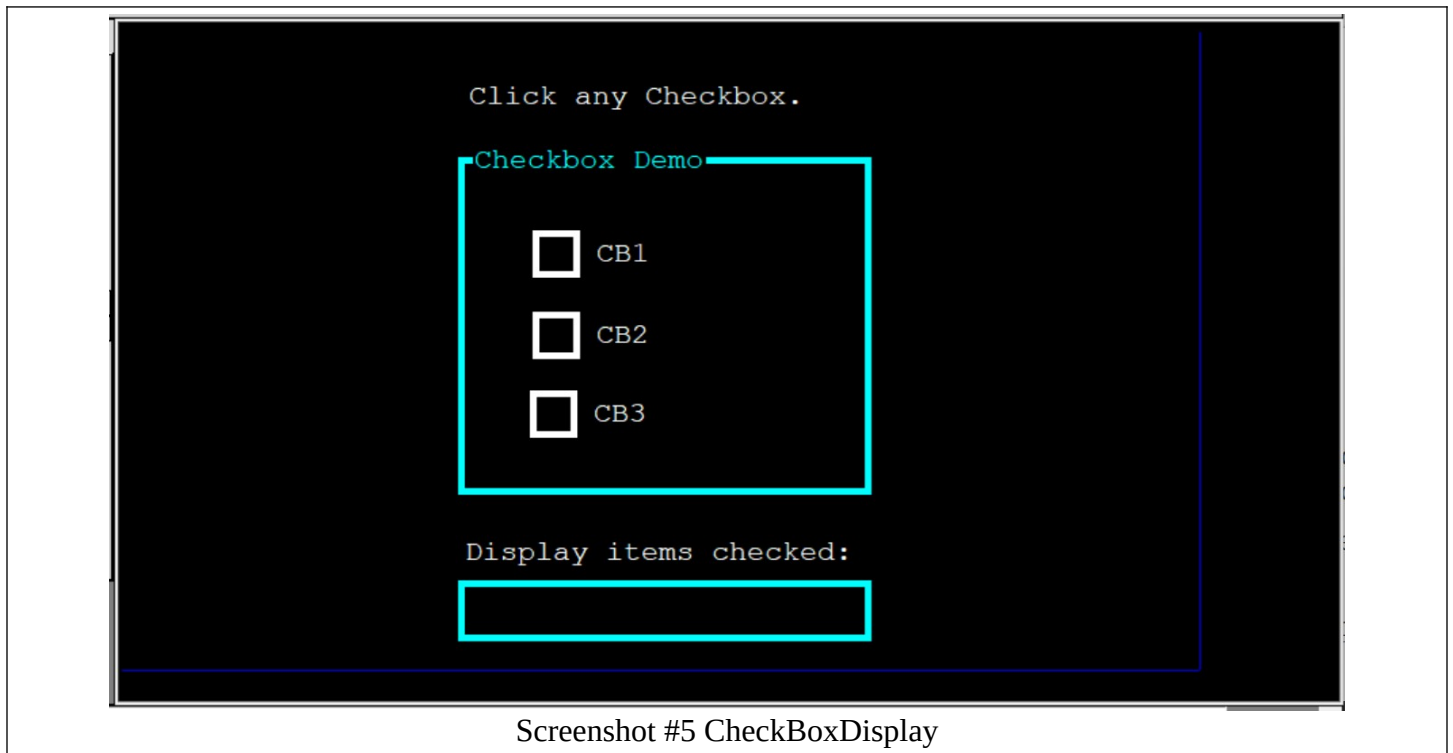
When ready the BAS can be opened in the MMBasic utilities (MMEdit5/MCC) as shown in Screenshot #3.



Screenshot #4 MMEdit5 and Maximite Control Centre

PDLB Tutorial

CheckBox Display Example



A. New Workspace and Project

After a fresh install, the workspace folder is specified. Using the Workspace wizard, click on New Folder and enter a name (for example: **WorkspacePDLB**). Select the folder and click the “Select Folder” button.

Click “New Project” to start the project wizard. Using the Project wizard, click on New Folder and enter a name (for example: **Tutorial**). Select the folder and click the “Select Folder” button.

Move the cursor over the Canvas area to update the Info Strip. The top two lines display the Workspace and Project Folder names.

Click the “Save Source File As” menu item, and use the File Wizard to save: **CheckBoxDisplay.src**

B. Place Controls

There are seven controls on the display page: Frame, three CheckBoxes, a DisplayBox and two Captions.

Using screenshot #5 as a guide, drag the controls and place them on the canvas as shown, and enter their captions. The CheckBoxes all have Default property “0”. The DisplayBox Default is “msg\$”. Using notepad,

create a text file called “**CheckBoxDisplayCode.txt**”. Enter this name into the Global Properties ”Code” field. Place the text file in the same folder as the “src” file.

Enter into the text file (note: the comment lines are single quotes) :

```
' initialize
Sub INIT()
End Sub

' main code
Sub MAIN()
  msg$ = ""
  if CtrlVal(#2) = 1 then msg$ = msg$ + "CB1 "
  if CtrlVal(#3) = 1 then msg$ = msg$ + "CB2 "
  if CtrlVal(#4) = 1 then msg$ = msg$ + "CB3 "
End Sub
```

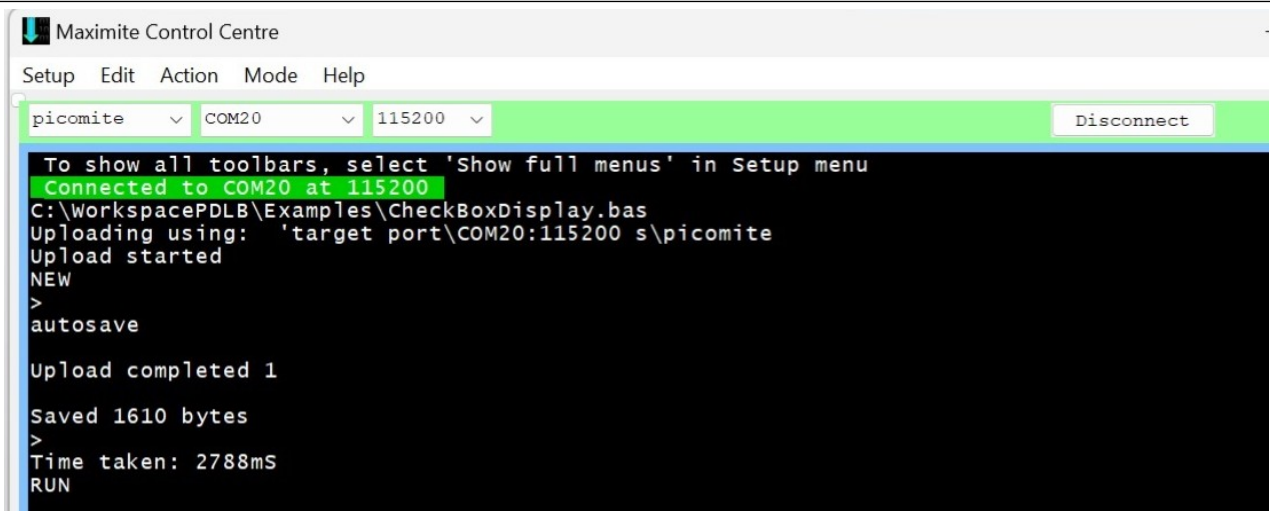
Code #2

C. Deploy

Click the Generate menu item “Build Controls “BAS” with Code” this writes the BAS text file into the Project folder.

Click the Generate menu item “Invoke Addon” which will push the current BAS file to MMedit5. You could make further changes or Press Function F2 on the keyboard. This will deploy(**) the code to the PicoMite.

****Note:** the Picomite platform should already be connected to the Desktop computer. The Maximite Control Centre should already be connected to the PicoMite (see Screenshot #6).



Screenshot #6 MCC – Maximite Control Centre

PDLB Main Menu Reference

File

Open Source	Opens the source file listed in the InfoStrip. If filename is blank it will attempt to open NewFile.src.
Save Source	Saves the source file listed in the InfoStrip. If filename is blank it will attempt to save any controls to NewFile.src.
Open Source File from	Using the file wizard, you can view the “src” files in the current project folder and select one to open. It ignores files in other folders.
Save Source File As	Using the file wizard, you can view the “src” files in the current project folder and enter a new filename to save.
Close Source	Clears canvas of all Controls. Like a reset.
New Project Folder	<p>Displays folder wizard to create new Folder in the workspace. Make sure you select Workspace folder, then click “New Folder”. It updates the current Project name in the Info Strip.</p> <p>Note: This feature also creates a subfolder called images and copies the “Cross.jpg” image into it (which is the default image.) It also identifies the project folder with a “builder_save.prj” file.</p>
Open Project Folder	Displays the folder wizard to allow you to select a previously created Project folder. It updates the current Project name in the Info Strip.
Create Workspace Folder	<p>Displays folder wizard to create new workspace folder. Make sure you select parent folder, then click “New Folder”. It updates the current Workspace name in the Info Strip.</p> <p>Note: This feature also creates a subfolder called images and copies the “Cross.jpg” image into it (which is the default image.) It also identifies the workspace folder with a special file in the distribution kit’s bin folder.</p>
Show Workspace in Command Prompt	If the pdlb.exe is invoked from a command prompt then you can use this command to see the current workspace foldername.
Open “Set” Workspace	Displays the folder wizard to allow you to select a previously created Workspace

	folder. It updates the current Workspace name in the Info Strip.
--	--

Edit

Renumber Controls	Should you edit the SRC manually you can use this command to renumber the controls.
Delete Selected	Select a Control on the canvas. You can delete it.

Font

1-8	PicoMite Fonts. Their pixel dimensions and (xxx) are depicted. Where: XXX indicates if all ASCII characters are part of the font. See PicoMite User Guide
Refresh fontsize globally	Routine will cycle through all controls looking for Text or Caption labels and it will change the fontsize. This action attempts to show the display as it will appear on the PicoMite

View

Show Grid	This puts a grid on the canvas to help you layout the controls. Default: “Dull” blue line marks the x=800 pixel and y=480 pixel boundary for the LCD Display.
Foreground Color	This sets the Colour command with the desired “default” foreground color. Default: white.
Background Color	This sets the Colour command with the desired “default” background color. Default: black.
Background Black	This sets black for the Colour command with the desired “default” background color.
Remove Grid	This removes the grid on the canvas.

Generate

Build Controls BAS Only	Writes a file with the ending “BAS” and contains the MMBASIC graphic Control commands.
Build Controls BAS With Code	Writes a file with the ending “BAS” and contains the MMBASIC graphic Control commands, and Control Defaults before the “Do-Loop” and Control Values inside the “Do-Loop”. If there is a Code file name in the Code field of the Global Properties. This command will merge the two sources by placing an INIT() before the “Do-Loop” and a MAIN() inside the “Do-Loop”. Init() and MAIN should be defined in your Code text file.
View output “BAS” file	This action will open the “BAS” file using the application in the Note field of Global properties. Default: Notepad.exe in Global properties causes this to view the BAS file with Notepad.
Invoke Addon	This action will open the Target file using the application in the Addon field of Global properties. Default: MMedt5.exe in Global properties causes this to view the BAS file inside the PicoMite MMEdit5 utility. Global Properties defines an Addon Target as (None, SRC, or BAS). Default: BAS

Help

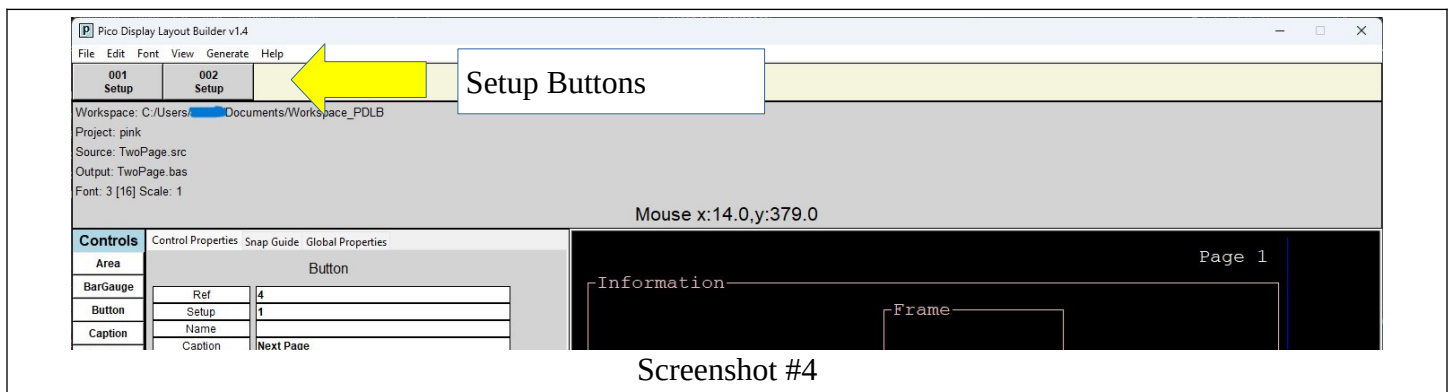
About	This action displays the copyright of the application and the current versions the application was built.
-------	---

Miscellaneous

PicoMite “Setup”

Control Objects can be grouped into a PicoMite “Setup”. When the designer adds a new Setup Control, all of the current Controls are hidden. New Control placements are associated with the new Setup page.

In Screenshot #4 below, two Setup groups are evident, by the Setup 1 and Setup 2 buttons in the Button Strip. The picture shows that the designer returned to Setup page one. Upon this action, the Setup page two objects are hidden, and the page one Controls re-appear. The designer can add more Controls to Setup page one.



Snap Guide

This feature provides alignment help to the Designer when they need to layout multiple controls on the Canvas. It is suggested that the “Show Grid” feature is also enablex.

This feature may require practice. Remember to enable a guide, use it, then disable the guide. Since there is no undo, you will have to re-select any inadvertent control moves and re-positioned them.

Essentially the guide works as follows:

- 1) Suppose you set guide x to 50. this is shown as a yellow line on the canvas.
- 2) Now any control you place (or touch) will set its StartX=50 and StartY= yyyy where yyyy is wherever you clicked. This does a pleasant “snap” to the x=50 pixel location.
- 3) Click any other control. And it too will “snap” to x=50. This action “aligns these controls”.
- 4) Make sure to disable the guide. Though Show Grid can stay enabled as it is passive only.

Platform

The Pico Display Base Board ([link](#)) is designed to interface to a five-inch SSD1963 LCD Display and many of the devices supported by PicoMite. The Base Board PCB was kept small and simple to make it affordable for multiple projects.

The PicoMite software includes an 8-bit parallel interface for the LCD display and SPI pins for the integrated Thin Film Transistor (TFT) Touchscreen and SD Card. The SSD1963 display boards all have a “standard” 40-pin male header, and so the desired display board is just inserted into the matching 40-pin female port.

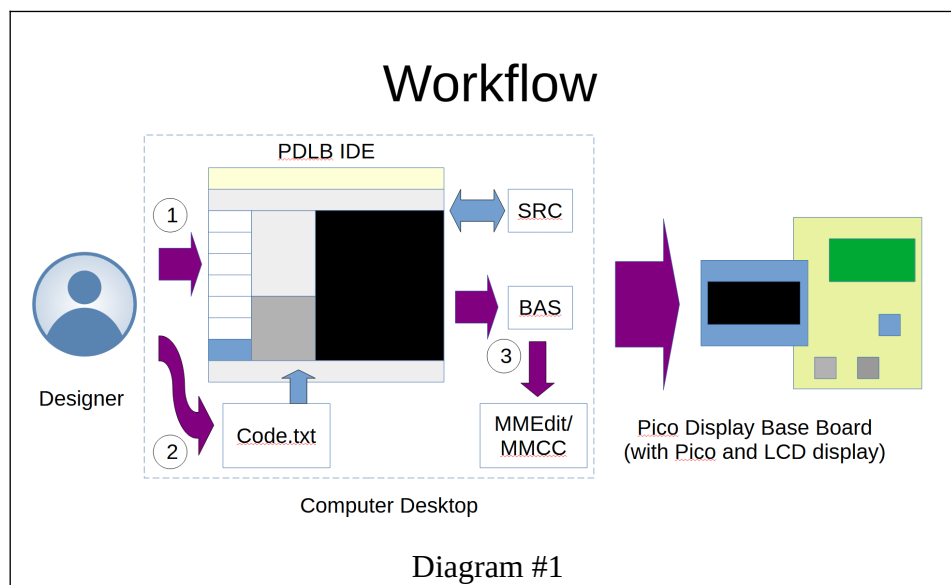
The base board expects a 5-inch LCD (800*480 pixels) with a touchscreen. The 40-pin male header is already soldered to the board. The LCD Display can be powered from the Pico. It also has an integrated SD Card which is supported by the PicoMite firmware.

PicoMite firmware is easily deployed to the Raspberry Pi Pico board. It includes MMBasic with advanced graphics and GPIO processing. A very good description of PicoMite is described by the author ([link](#)) The PicoMite authors also released MM Edit and MM Console (which is excellent software to edit/deploy the code from a desktop computer).

Managing Source and Deployment

Since the PicoMite on Pico is so versatile, a designer will want to create multiple projects. The PDLB assists the user by creating Workspace and Project folders. Within the workspace, the designer can create one or more projects. Within the project folder are the image, source “SRC”, and Basic “BAS” files. The names of the current workspace, project and source files are displayed on the IDE’s Info Strip.

The suggested workflow (see Diagram #1) is to (1) use PDLB to layout the Controls and generate a “BAS” file. (2) And, include an optional code file. (3) use MM Edit to open the “BAS” file and modify the code further. Click the “Deploy” button. This automates the transfer via MM Console to the Pico and executes the code.



Alternately, one can edit the “BAS” file and then copy/paste directly into PicoMite using TeraTerm’s terminal.ⁱ

i Raspberry Pi Pico (Pico) is a product of Raspberry Pi Ltd.
PicoMite is a product of Geoff Graham and others. Visit <http://www.thebackshed.com>
MMEdit, and MMConsole are products of mmedit@c-com.com.au
TeraTerm is a product of the Tera Term Project. See <http://ttssh2.sourceforge.jp/>
Tindie is a product of tindiw.org. See <https://tindie.com>
Pico Display Base Board is a product of KW Services. See his Tindie store.