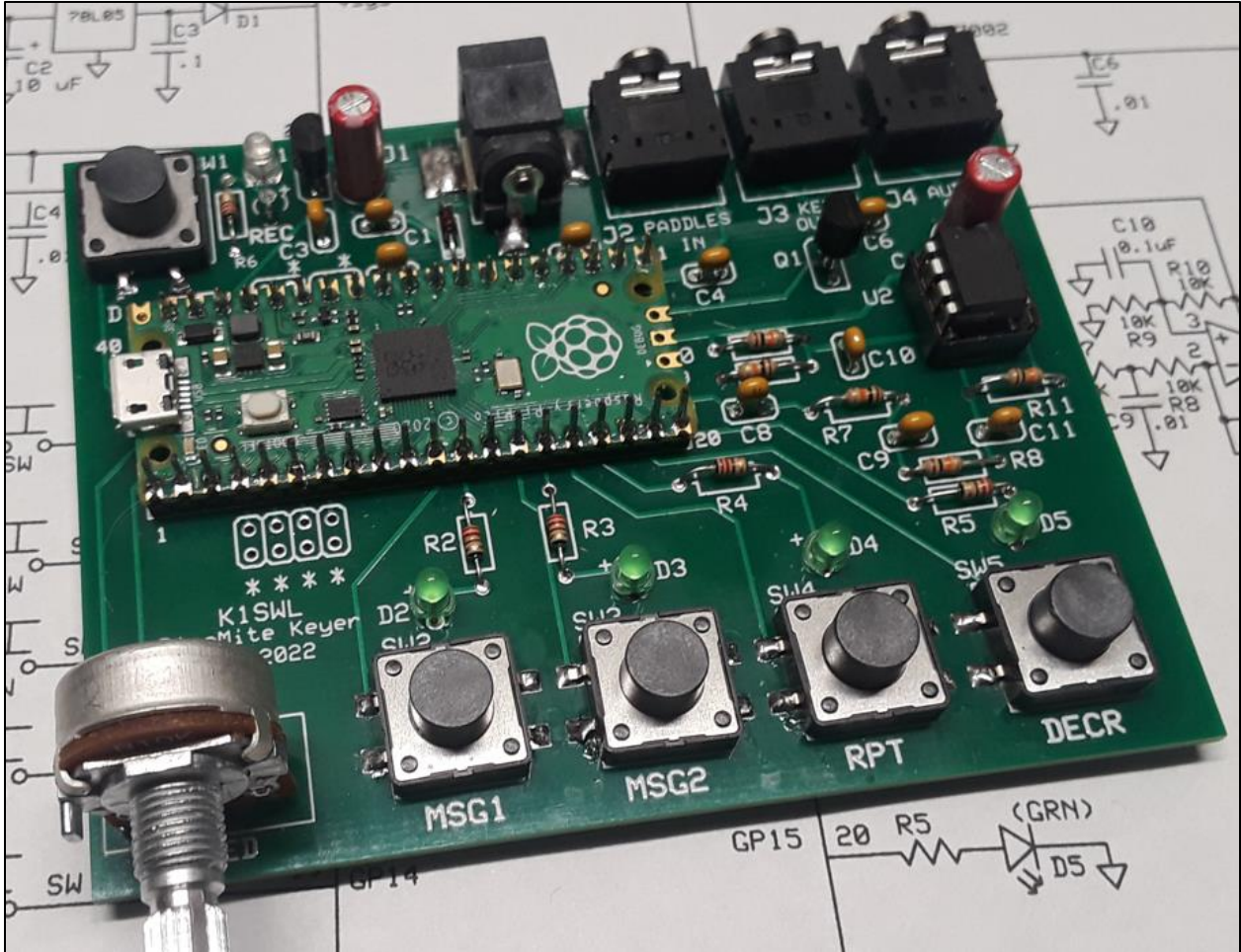


PicoMite Memory Keyer (PMK)

Assembly & Setup Instructions



Welcome aboard, and thanks for your interest in the PMK Keyer!

Assembly order is fairly non-critical, but I'll note anything of interest along the way.

It's a good idea to review your collection of parts before launching into assembly. If you find that parts are missing, contact George (n2apb@MidnightDesignSolutions.com) for prompt assistance.

Static-sensitive parts

The ICs and the Pico may be considered static-sensitive. If possible, ground yourself while installing them. A grounded wrist strap is ideal. If none on hand, touch something grounded frequently while working. Avoid moving around on carpeting and also avoid resting these components on paper.

'Tight spaces'

There are a few. The components immediately above the Pico are 'down in the hole' if you leave them toward the end of assembly. That's true of R12 and R13 (below J2) and the 2-pin header between J1 and J2 as well.

Polarity-sensitive Components

The two 1N4148 diodes (D1 and D7)

Ensure that the banded ends of the diodes match that shown on the silkscreen for those devices.

The two electrolytic Caps (C2 and C12)

The longer of two wire leads is positive (+). Ensure that it's installed with that lead closer to the back edge of the PCB. This puts the vertical band on the cap toward the front of the board.

The five LED indicators (D2- D6)

The longer of two wire leads is positive (+). Ensure that it's installed with that lead closer to the + sign on the silkscreen.

The two integrated circuits (U1 and U2) and Q1.

Ensure that the 'flats' on U1 and Q1 face in the direction shown on the silkscreen.

Ensure that the notch on U2 faces 'up' (toward C12). *Note: if you install the socket upside down, leave it. It too has a notch, but that doesn't affect operation in either orientation.*

The Pico itself

The Pico **must** be installed with the USB micro connector facing the Left edge of the board.

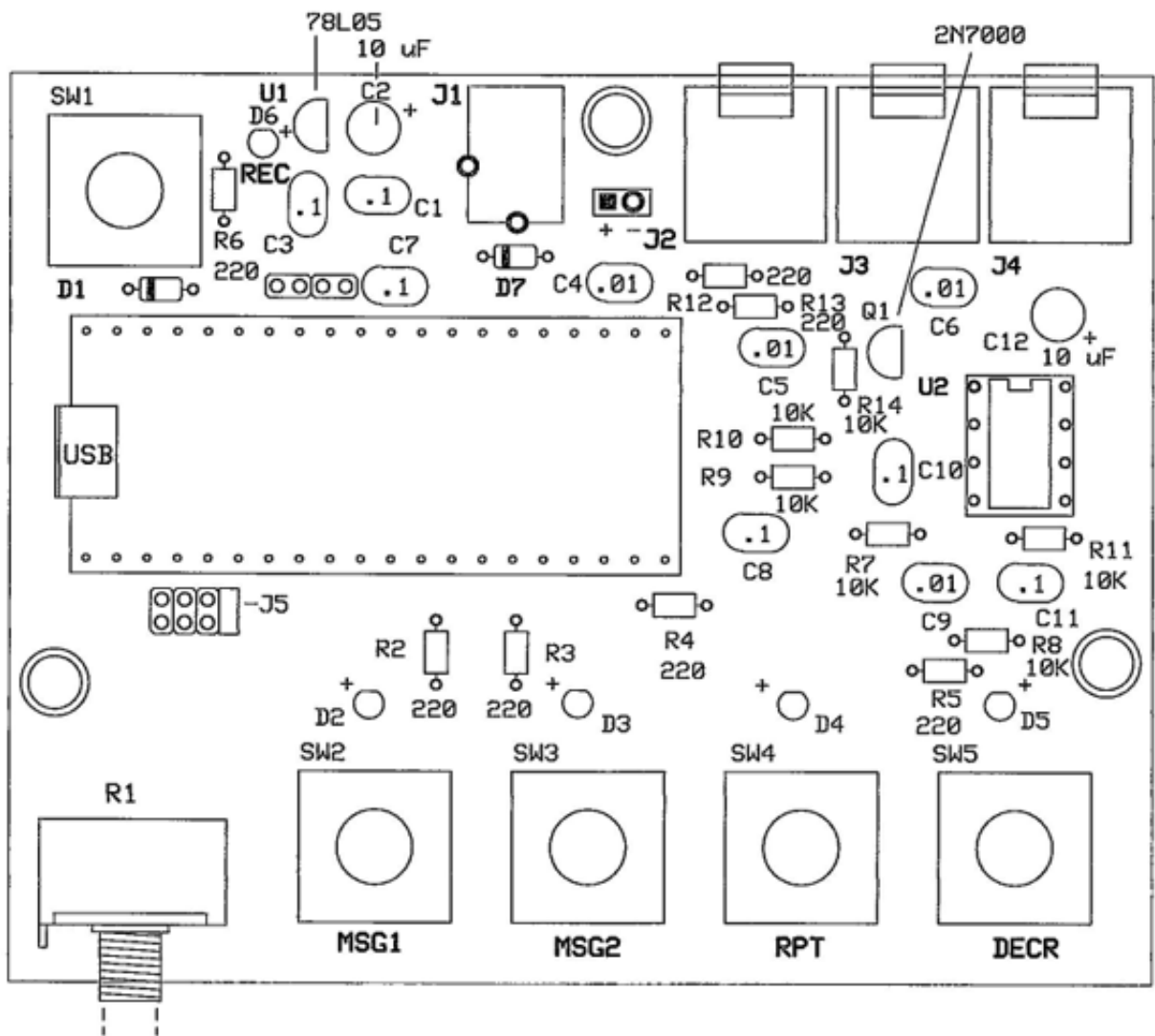
The Pico installation is the only tricky step in the assembly process.

- 1) Install the two 20-pin headers on the Pico board, but **do not solder yet**.
- 2) Likewise install the two 20-pin headers on the main circuit board. **Do not solder yet**.
- 3) Mate the headers and sockets together on the board. You'll need to keep the assembly secure with a fingertip.
- 4) Tack at least one pin on the Pico with solder. Invert the board and solder at least one socket pin to the underside of the board. This is a 3-handed operation. I hang a length of solder over the edge of my bench and maneuver the board to the solder. Once this is achieved, solder all remaining connections on the Pico and the socket strips.

Here's why: if the sockets or header pins are tipped, you won't be able to install the Pico in the sockets. Ask me know I know this.

SMTs: There are five of them – the tactile switches. They're huge by SMT standards.

Add solder to one of the four SMT pads. Hold the switch in place while heating that corner. Once cooled, proceed to solder the remaining 3 pads on each switch. *Easy-peasy.*



Software Installation

The software installation process for the PicoMite is straightforward and the website <https://geoffg.net/picomite.html> describes the operating environment and features.

Step 1: Download the MMBASIC software, the Manual and the .uf2 Boot Loader

Toward the bottom of Geoff's page is a dialog button for downloading the MMBASIC software. It's in zipped form, and the .uf2 file therein contains the MMBASIC code. The companion document - the User Manual - is included in that zipped directory and describes the installation process in detail on pages 5 and 6. That's not all - the manual runs 172 pages in all and is an excellent and comprehensive guide to MMBASIC. Unzip the file to a folder located in a convenient place on your hard drive.

Step 2: Install the Boot Loader

- a) Connect the Pico to your computer using a USB cable ... while at the same time holding down the small white tactile switch on the Pico. This action will trigger the computer to open a new folder in the Windows File Manager.
- b) Locate the .uf2 file (from step 1 above) and drag it into this newly created folder.
- c) The Pico will immediately reboot and appear as a new serial device as can be seen in Windows Device Manager. Its LED will be blinking to indicate that MMBasic is loaded and running!

Step 3: Load a Terminal Emulator to Windows to Communicate with the Pico

We recommend locating and loading the 'TeraTerm' terminal emulator app to your computer. It is free and quite usable. <<https://osdn.net/projects/ttssh2/downloads/74780/teraterm-4.106.exe/>>

Once installed and started, click the 'Serial' radio button, and see the standard COM1 and the newly-created COM port (e.g., COM3, COM7, etc). Select the latter and hit carriage return. You should see a command prompt ('>') appear on the terminal. You're now running MMBasic!

The installation took about fifteen minutes, and most of that time was spent on reading up on the installation steps. The process itself went flawlessly on a Windows 10 machine, and versions are also available for MAC and Linux. The software driver necessary for Windows 7/8 isn't supported by Microsoft, but a link to a generic driver is provided on page 7 of the User Manual. This too was successfully installed on my 'workhorse' Windows 7 computer.

Step 4: Load the PMK Software

This last step is to download and install the PMK software. Currently at version 2.0, it is located on the PMK website as a text file. Right-click on the link and **Save As ...** to that same convenient place on your computer's hard drive.

Use the TeraTerm's "XMODEM" command to transfer the program file to the Pico, and the PicoMite's Xmodem Receive command to gobble up the PMK software being sent to it by the computer.

- a) At the TeraTerm command prompt, type: xmodem receive
- b) In the TeraTerm menus along the top, select: FILE..Transfer..XMODEM..Send, then navigate to and select the PMK software you just downloaded.
- c) When the command prompt '>' reappears, the PMK software is will be loaded successfully into the Pico's nonvolatile program memory space. (Be sure to read in the MMBasic manual about the seven other memory banks available on the Pico into which you may transfer programs in/out for storage and retrieval ... nice for saving backup copies on-hand when making changes!)

Setup Control 'OPTIONS'

There are just a few **OPTIONS** that should be setup first thing for the Pico, as a matter of convenience. These following commands are issued only once – they stay in place as long as MMBasic remains loaded on the device.

- Option Colourcode ON
- Option Display 60,100
- Option System i2c GP2,GP3
- Option LCDPANEL SSD1306i2c,RL
- Option Audio GP20,GP21
- Option Autorun ON
- Option TAB 3

Two Operating Modes for the PMK

The PMK has two distinct operating modes: 'Terminal' and 'Standalone'

Terminal – In this mode one relies on terminal running on your computer for setup and message entry. Upon connecting the USB cable, the PMK software comes alive with a user dialog. It prompts the user to enter two messages and make a selection of three other features. These are 'Reset Serial number', Paddle reversal and Iambic Mode A/B.

Standalone – In this mode only the keyer paddles are needed to initialize the PMK functions. Pressing the 'Record' button arms the programming mode for entering a new message. At the end of the message, press one of the four message pushbuttons to store it in the desired location and the Record LED will turn off. Press that message buffer button to hear your newly stored message! , This sequence is shorter since the user won't change the keyer settings often if at all.

The Development Process

It's very convenient and quite fun to make changes to this open source PMK code. "Open source" basically means that you are quite welcome to make changes to the code for your own personal use, but you are more than encouraged to share those changes back with the PMK development community. We have the PMK v2.0 code on GitHub and it's easy to create your own branch of the source containing your changes so others can benefit by them. The official code is at <https://github.com/n2apb/pmk>

From the command prompt '>', pressing your computer's 'F2' button starts program execution. 'Control-C' halts execution. From the command prompt, typing 'EDIT' or pressing 'F4' brings up a very nice and colorful editor. Its error messages seem considerably better than the compiler errors thrown by the Arduino environment. As an interpreter-based software, it tells you exactly where the problem lies. Once your changes are made, exit the editor and run the code straight away!

Other Notes

The PMK retains the last serial number if the PMK is not reset. This allows the keyer to recover gracefully in the event of a power interruption. *All operational settings including the messages are stored in non-volatile memory and are restored upon application of power.*

We'll likely expand this manual with description of the software and various ways you can expand/modify it for your own use.

Meanwhile be sure to let us know what you think!

73-

Dave Benson, K1SWL, email: davek1swl@gmail.com

George Heron, N2APB, email: n2apb@MidnightDesignSolutions.com