Game★Mite

User Manual

Revision 1.2.1

by Tom Williams, Mick Gulovsen & Bill McKinley

Table of Contents

Revision History

1.0.0	Initial release for GameMite firmware/software v1.0.x and PCB v1.3.x.
1.0.1	Fix descriptions of right-angle headers and add missing headers for mounting modules to BOM.
1.2.0	Updated for PicoMite v5.08 firmware and GameMite v1.2.0 software.
1.2.1	Titivated instructions regarding the optional "Charge jumper" (J9).

Copyright and Acknowledgements

1. Introduction

The GameMite is a "build it yourself" handheld game console based on the Raspberry Pi Pico microcontroller running the PicoMite firmware.

Features

- Raspberry Pi Pico Microcontroller
- 320x240 ILI9341 SPI TFT display + integrated SD card reader
- NES style game controller (very clicky)
- Mono audio via PWM capable of playing 4 notes simultaneously
- 3v7 LiPo battery with charging and protection circuit
- MMBasic programming language

Support

Support questions should be raised on "The Back Shed" forum (https://www.thebackshed.com/forum/Microcontrollers) where the developers of the GameMite and the PicoMite firmware and many other MMBasic users are regulars.

2. Building the GameMite

Construction of the GameMite is fairly straightforward and can be completed by those who have a basic knowledge of soldering through hole components, there are no surface mounted components to worry about.
That aside there are a few areas that can be a bit tricky, but if done in the order set out here you shouldn't have any real problems to complete the build.

Of course, the usual safety precautions should always be adhered to; wear glasses when trimming wires, and whilst soldering, and be careful handling the hot soldering iron.

See the bill of materials (Appendix B) and the overlay image below, for component orientation and values.

IMPORTANT: Two copies of the PCB are used in construction; the top one has the components soldered to it, the bottom one serves as a baseplate and somewhere to attach the speaker and battery. If you bought your GameMite as a kit then it is possible that the PCB supplied for use as the baseplate is an older revision. Check the version number (lower right hand corner) of both PCBs and **only solder components to the one with the higher version.**
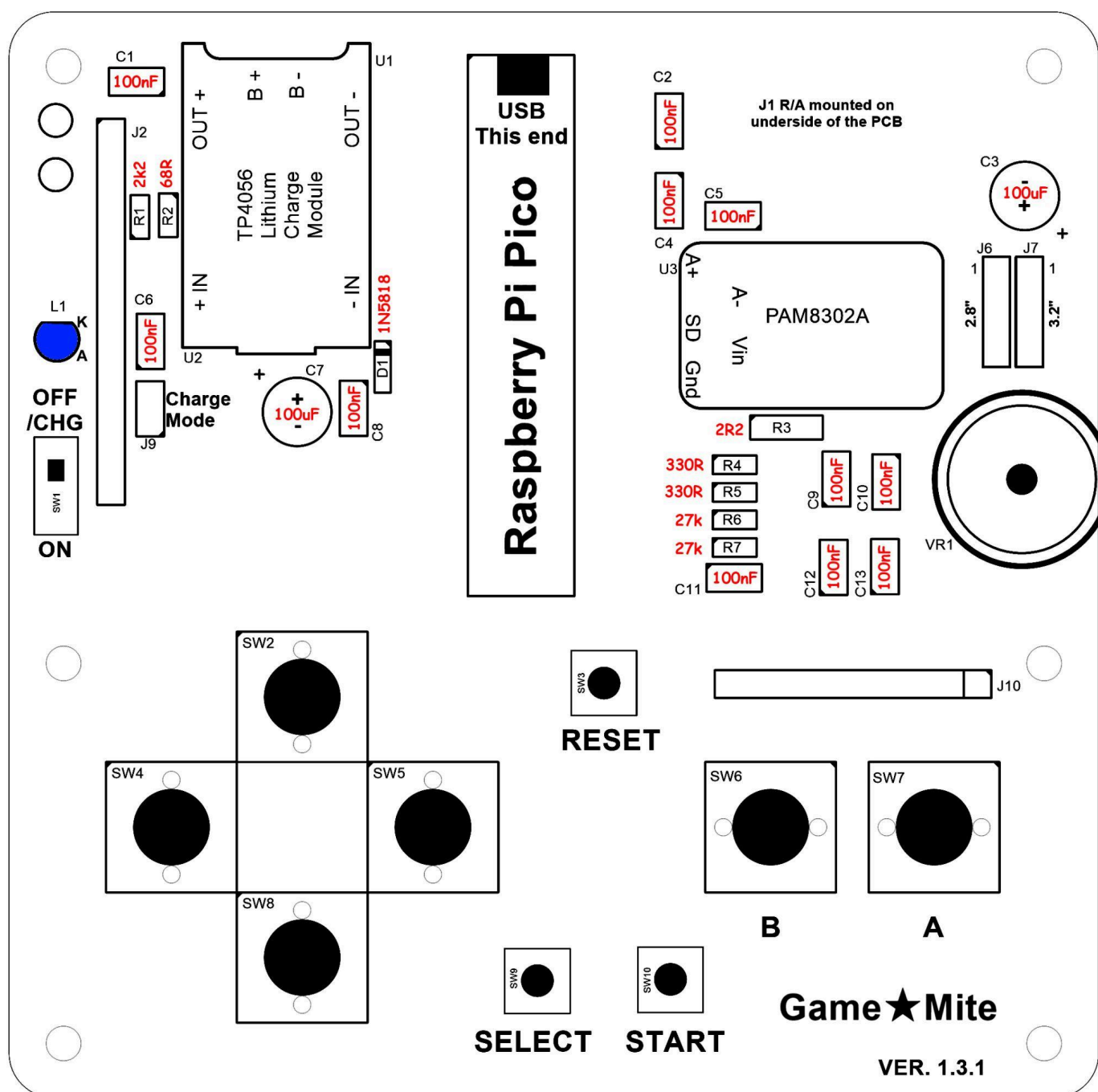
A note about the modules

If you don't read any further, make sure you read this paragraph!

The GameMite uses 3 modules as part of its build, namely: The Raspberry Pi Pico, TP4056 lithium manager and a PAM8302A amplifier. It is strongly recommended that all three modules should be spaced off the PCB by soldering them onto standard male header pins.

- **DO NOT** socket the modules as there is insufficient room under the TFT.
- **DO NOT** directly solder the Pico to the PCB using its castellated pads as you then won't be able to insert a micro USB cable.

The male pins

Normally, construction would begin with the passive components (resistors, capacitors etc.) but several components are on their own modules with pin spacings that are not all 2.54mm (0.1"). As a result individual pins have to be cut from strips and soldered onto the PCB, getting them all aligned perfectly square can be a bit awkward.

One way to achieve this is to use a second PCB as a jig:

1. Insert male pins into the PCB you are assembling, long side down from the top, in the locations shown in RED .
2. Place a second PCB over the top of the short side of these pins sandwiching them between the two PCBs.
3. Use small clamps, clothes pegs, or elastic bands to hold the assembly together.
4. Flip the assembly over and solder the pins to the underside of the **first PCB**.
5. Once soldered, **carefully** snip off the excess pin length from the underside of the first PCB.
6. Remove the second PCB.
7. You should be left with the pins, marked in RED, with their short side facing UP. You will solder the modules to these later in the build.
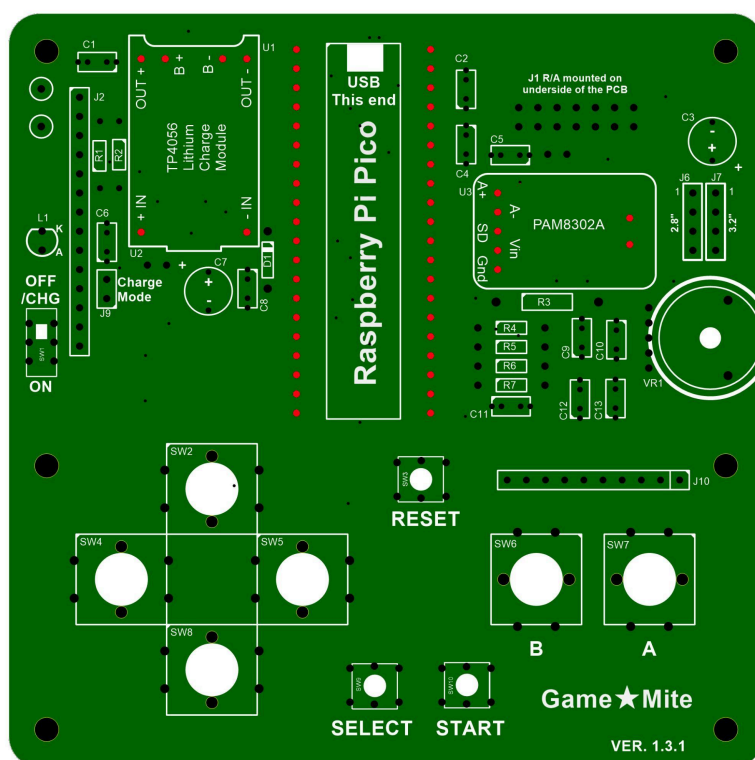
Solder side of the higher version PCB with long pins protruding

Second PCB to help locate the pins for soldering

Revision 1.2.1
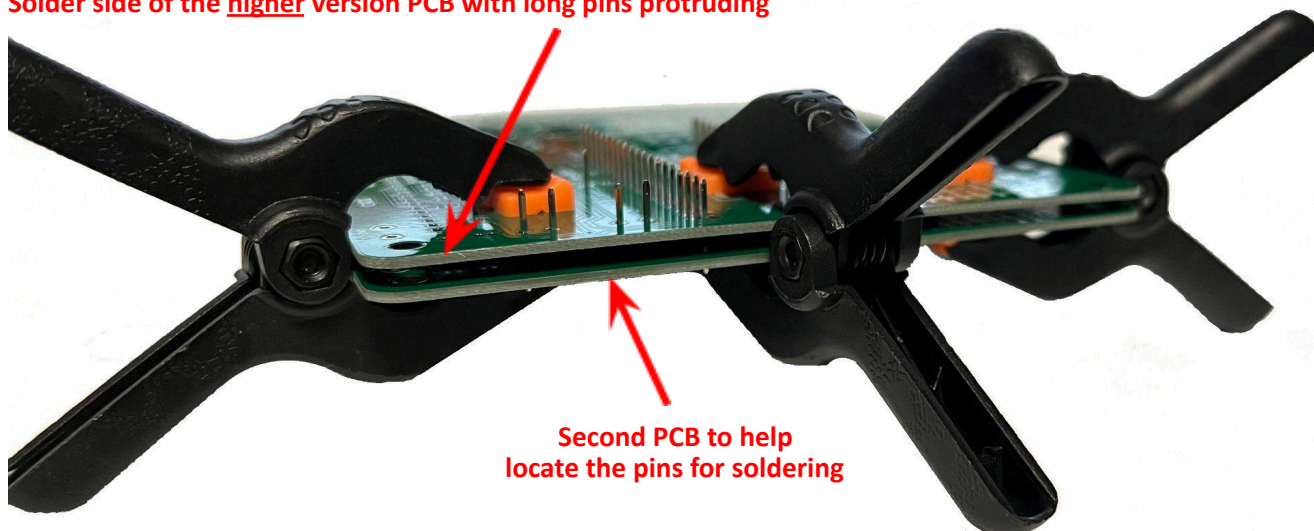
Resistors and capacitors

Refer to the bill of materials (Appendix B) and the overlay for the value of each component and solder them to the GameMite, using the placement that is silk screen printed onto the PCB.

Fit the resistors, 100nF capacitors (the PCB caters for both 2.54mm/0.1" and 5.08mm/0.2'' spacing) and the 1N5818 diode (D1).

DO NOT fit the 2 electrolytic capacitors (C3 & C7) at this time. This is because they can make it more difficult to fit the speaker (J5) and battery (J8) sockets, see next section.

Fit the volume control (thumbwheel trimpot VR1), charging-indicator LED (L1, observe the polarity; the anode is the longer leg and the cathode has a small flat on the edge) and all of the switches (SW1-10).

Fit J2 and then depending on your TFT module size fit J6 (for 2.8" TFT) or J7 (for 3.2" TFT); only one of these is required but it won't hurt to install both.

Optionally, for test purposes, fit a small half loop of bare wire connecting the two GND test points at the top left of the PCB to provide an attachment point for a crocodile clip.

DO NOT fit anything in the J9 or J10 positions:

- J9 is for an optional "Charge mode" jumper, see Appendix D - "Powering the GameMite".
- J10 allows an alternative controller scheme to be fitted on a daughter board.

Fit the modules: TP4056 (with the USB, which is unused, pointing towards the bottom of the PCB), Raspberry Pi Pico (with the USB pointing towards the top of the PCB) and the PAM8302A into their respective locations and solder them in place.

Use a screwdriver with a small flat blade to carefully turn the miniature trimpot on the PAM8302A module fully clockwise (maximum resistance). **DO NOT** use a Phillips / cross-head screwdriver; despite appearances to the contrary the screw is not of that type and you will possibly chew up the head. See Appendix D for more details.

Optionally, fit the expansion connector J1 onto the **underside** of the PCB.

Connecting the speaker and battery

The speaker (J5) and battery (J8) are connected to the **underside** of the top PCB by flying leads:

- **BE CAREFUL** to correctly observe the polarity of the battery.
- **DO NOT** short the battery leads.
- **DO NOT** swap the speaker and battery.

For the battery we recommend that a polarised connector is used, e.g. on the PCB a right-angle 2-pin JST-XH socket.

For the speaker an unpolarised connector is sufficient, e.g. on the PCB a right-angle female 2-pin 2.54mm pitch standard header.

The flying leads for the battery and speaker will then need the appropriate mates spliced to them with heat shrink tubing or insulation tape applied.

Other connection methods are possible: the pin spacing for both connectors is 2.54mm and will accept standard right-angle male or female headers or small screw terminal blocks; at a pinch you could even directly solder the leads to the PCB.
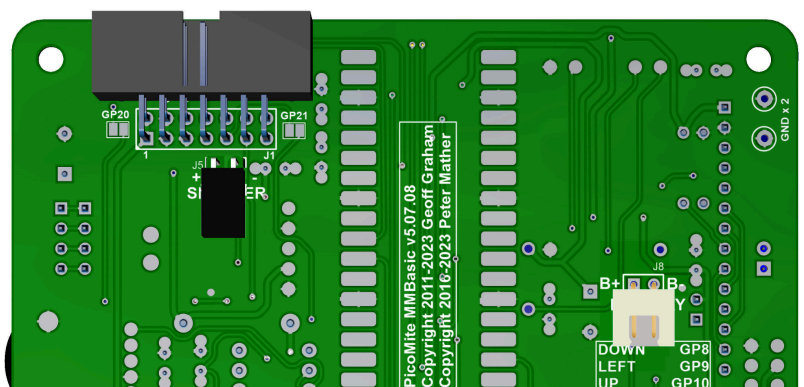
Finalising the build

Fit the 2 electrolytic capacitors (C3 & C7). Observe the polarity and mount them rocked over on their side (see photo of constructed board) to clear the TFT screen when it is fitted (there is 11mm clearance under the TFT).

Fit keycaps onto the 12mm switches then connect the battery, speaker and choice of 2.8" or 3.2" ILI9341 TFT SPI display.

Secure battery and speaker to the top surface of the baseplate PCB, using double sided foam tape or other suitable mounting mechanism.

> **WARNING!** Be very careful to ensure that the pins on the underside of the board are sufficiently trimmed so that the battery will not be pierced by any of them!

Finally mount the main PCB to the baseplate using 6 x 10-13mm (0.4-0.5") standoffs (fitting just the 4 corners should be sufficient).
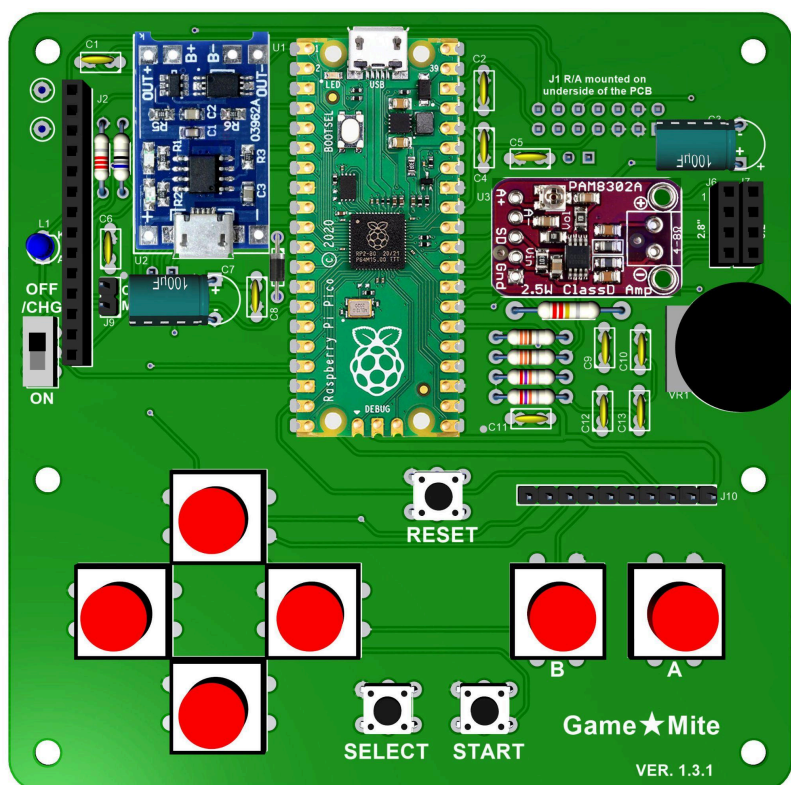
Charging the battery

With SW1 up (OFF/CHG) plug a USB power source into the Raspberry Pi Pico. The 3mm blue LED (L1) will light to indicate that the GameMite is in charge mode; note that this stays lit even when the battery is fully charged. In addition there is a small red LED on the TP4056 which will light whilst the battery charging, once it is fully charged this will be replaced by a small blue LED.

Once the battery has been charged for the first time see Chapter 3 to install the firmware and software.

3. Installing the GameMite firmware and software

WARNING! The installation process described here will replace the entire contents of the Raspberry Pi Pico's flash memory **erasing any existing content**.

The simplest way to install the GameMite software uses a single file image (.uf2 file) containing both an appropriately configured copy of the PicoMite firmware plus the GameMite software (menu, tools and games.)

Note: see Appendix C for an alternative installation process that allows the PicoMite firmware and GameMite software to be installed and updated independently of each other.

To install:

1. Follow the links from https://github.com/thwill1000/mmbasic-gamemite#firmware-and-software to download the latest "GameMite-<version>.uf2.zip".

2. Extract the "GameMite-<version>.uf2" file.

3. Using a USB cable, plug the Raspberry Pi Pico into your computer (Windows, Linux or Mac) while holding down the white BOOTSEL button on the Raspberry Pi Pico (you will need to remove the TFT display to do this).

4. The Raspberry Pi Pico should connect to your computer and create a virtual drive (the same as if you had plugged in a USB memory stick) called "RPI-RP2". This drive will contain two files which you can ignore.

5. Copy the "GameMite-<version>.uf2" file to this virtual drive.

6. When the copy has completed, the Raspberry Pi Pico will restart and create a virtual serial port on your computer. The LED on the Raspberry Pi Pico will blink slowly indicating that the firmware with MMBasic is now running.

   If the LED does not start blinking, then it is possible that the Raspberry Pi Pico's flash memory was in a previously corrupted state. To fix this install the file https://github.com/thwill1000/mmbasic-gamemite/raw/main/xtra/flash_nuke.uf2 following the same instructions. This should reset the Raspberry Pi Pico to its factory fresh state from where you can try installing "GameMite-<version>.uf2" again.

7. The GameMite software should now be set up. If RESET is pressed or the device is power-cycled it should reboot to show the GameMite splash-screen before displaying the menu.

4. Adding new programs to the GameMite menu

When the GameMite is started for the first time it will create the file "A:/.gm-menu" with contents similar to this:

```
Circle One,circle.bas
Lazer Cycle,lazer-cycle.bas
PicoVaders,pico-vaders.bas
Yellow River Kingdom,kingdom.bas
3D Maze,3d-maze.bas
Controller Test,ctrl-demo-2.bas
Sound Test,sound-demo.bas
File Browser,fm.bas
```

This file can be edited (e.g. with a terminal emulator and the MMBasic EDIT command) to add or remove items from the menu.

The format of each item is:

```
<menu-text>,<file-path>
```

e.g.

```
PETSCII Robots,B:/mmbasic-robots/robots.bas
```

If a <file-path> is not absolute then the GameMite will look for the program first in the "A:/GameMite" directory. If the program is not not found there then it will look in the "B:/GameMite" directory.

If the file contains more items than can be displayed on a single screen then the GameMite menu will have multiple pages which can be navigated between using the LEFT and RIGHT buttons.

5. Programming the GameMite

From a programming aspect the GameMite hardware is just a standard PicoMite with an ILI9341 TFT display attached to it. As such, those who wish to write their own programs for it are directed to the excellent PicoMite User Manual (https://geoffg.net/Downloads/picomite/PicoMite_User_Manual.pdf).

The one non-standard (because there is no PicoMite standard) part of the hardware is the controller layout and how to read this is demonstrated in the section on "Reading the controller".

Determining if the current program is running on a GameMite

Assuming `OPTION PLATFORM` has been set correctly the following MMBasic snippet will identify whether a program is running on a GameMite:

```
If Choice(Mm.Device$ = "PicoMite", Mm.Info$(Platform) = "Game*Mite", 0) Then
  Print "This is a Game*Mite"
Else
  Print "This is NOT a Game*Mite"
EndIf
```

DO NOT do this:

```
If Mm.Device$ = "PicoMite" And Mm.Info$(Platform) = "Game*Mite" Then …
```

as it will report an `ERROR` on devices that do not support `OPTION PLATFORM` such as the Colour Maximite 2[1].

---

[1] This occurs because MMBasic does not perform short-circuit evaluation of boolean expressions and will try to evaluate the `Mm.Info$(Platform)` expression irrespective of the value of `Mm.Device$`.

Reading the controller

The following program demonstrates how to read the GameMite controller:

```
' simple_ctrl.bas

Cls

' Initialise GP8-GP15 as digital inputs with PullUp resistors; if the
attached
' button is pressed then the corresponding pin will read low (0), otherwise
it
' will read high (1).
For i% = 8 To 15
  SetPin Mm.Info(PinNo "GP" + Str$(i%)), Din, PullUp
Next

Do
  ' Read digital inputs into the bits of a single integer.
  ' We invert the raw GPIO input so that if a button is pressed the
  ' corresponding bit is set, otherwise it is unset. If multiple
  ' buttons are pressed simultaneously then multiple bits will be set.
  x% = (Port(GP8, 8) Xor &h7FFF) And &hFF

  ' Display buttons currently pressed
  s$ = ""
  If x% And &h01 Then Cat s$, "DOWN "
  If x% And &h02 Then Cat s$, "LEFT "
  If x% And &h04 Then Cat s$, "UP "
  If x% And &h08 Then Cat s$, "RIGHT "
  If x% And &h10 Then Cat s$, "SELECT "
  If x% And &h20 Then Cat s$, "START "
  If x% And &h40 Then Cat s$, "B "
  If x% And &h80 Then Cat s$, "A "
  If s$ = "" Then s$ = "-- NONE --"
  If Len(s$) < 40 Then Cat s$, Space$(40 - Len(s$))
  Text 0, 0, s$
Loop
```

Note: for consistency between GameMite programs it is recommended that the START button is used to start/pause programs and the SELECT button is used to select an item highlighted in a menu. In some instances it might make sense for the A button to start/pause a program or select an item, but this should be in addition to, not instead of the the START and SELECT buttons.
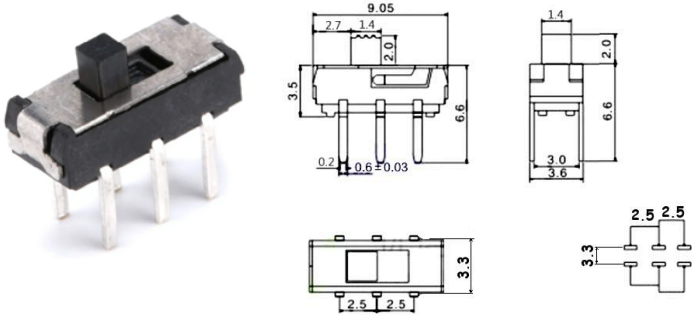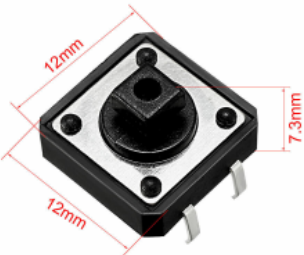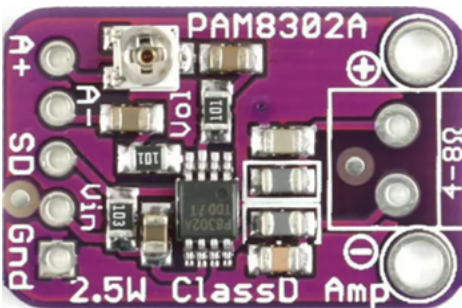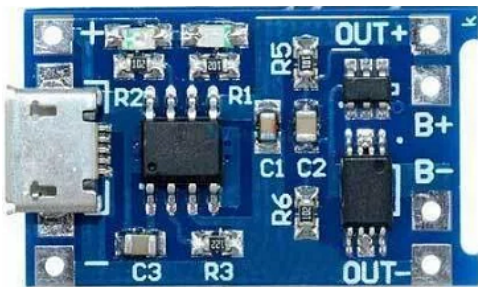
Appendix A - Schematic
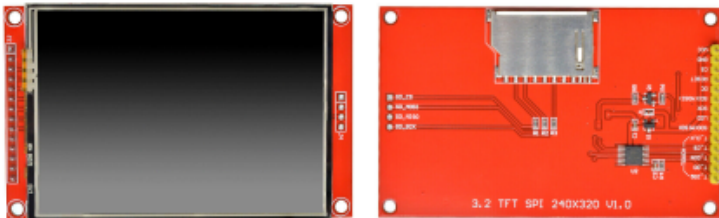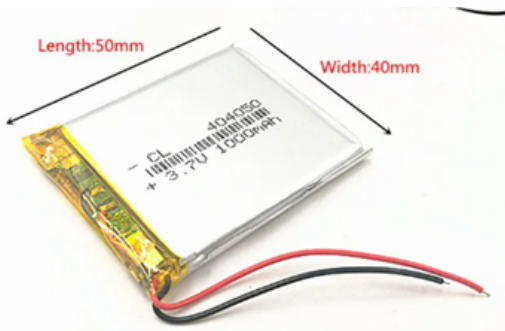

Game★Mite Schematic
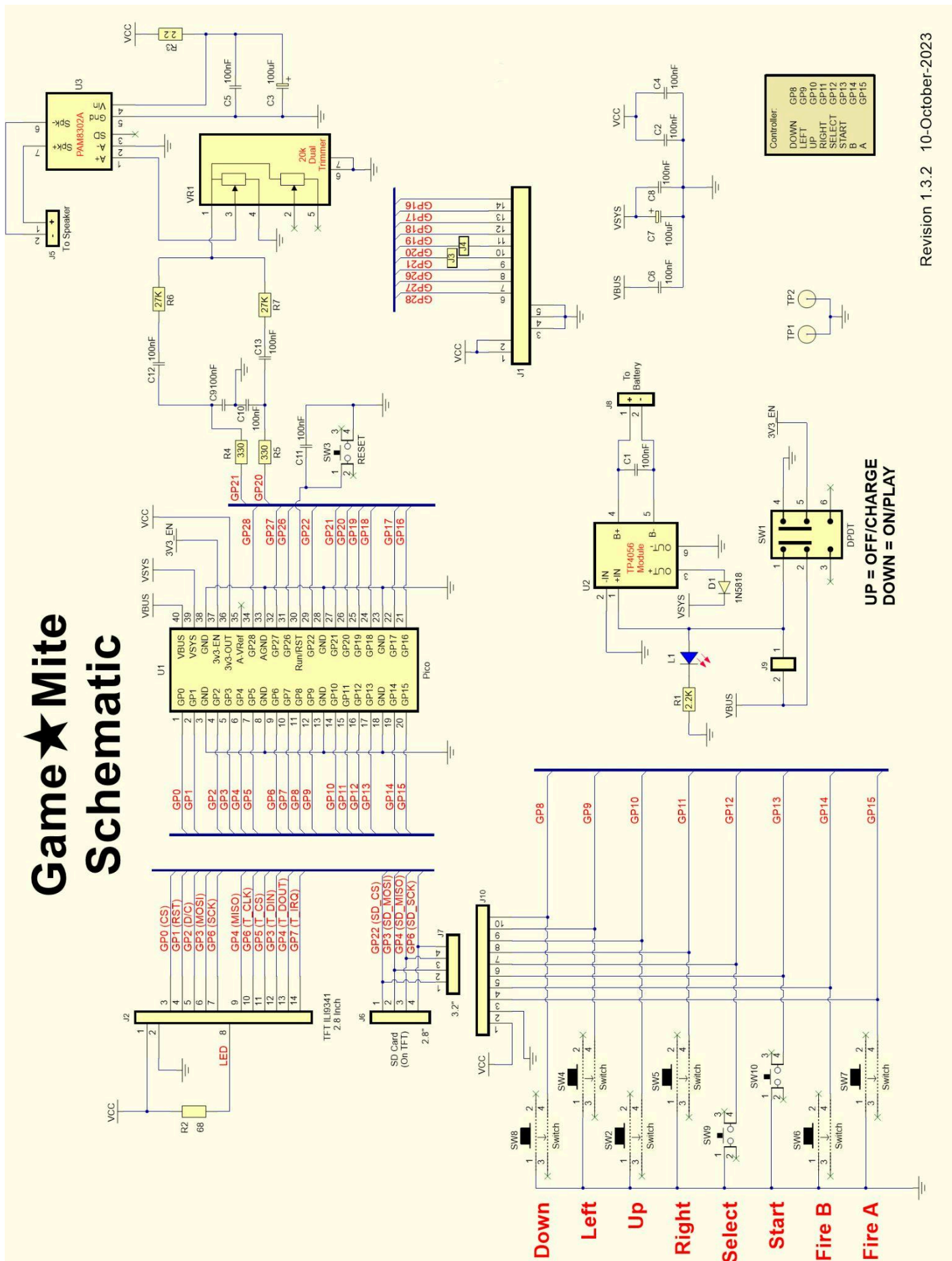
Revision 1.3.2   10-October-2023

Appendix B - Bill of materials

Because searching the internet based on a component description alone may point to an incorrect part we have included pictures below for some items. Items not matching the illustrations may not be compatible.

Count	Reference	Value	Comment
2	Game★Mite PCB	-	Two copies of the PCB are used in construction; the top one has the components soldered to it, the bottom one serves as a baseplate and somewhere to attach the speaker and battery. If you bought your GameMite as a kit then it is possible that the PCB supplied for use as the baseplate is an older revision. Check the version number (lower right hand corner) of both PCBs and **only solder components to the one with the higher version.**
11	C1,C2,C4,C5,C6,C8, C9,C10,C11,C12,C13	100nF	Mounting holes suit 0.1" or 0.2" spacing
2	C3,C7	100uF 10V	Mounting holes suits 0.2" spacing
1	D1	1N5818	Schottky diode
1	J1 (Optional)	Right-angled boxed header 7x2M	Expansion header exposing GP16-21,26-28
1	J2	Header 14x1F	Cut from larger strip
2	J3, J4	-	Solder short pads, close with a blob of solder to make GP20 + 21 available from J1
1	J5	Right-angled header 2x1F	Speaker "socket" soldered on PCB
1	-	Header 2x1M	Speaker "plug" on lead to match J5. Cut from a larger strip.
1	J6	Header 4x1F	For 2.8" display. Cut from larger strip
1	J7	Header 4x1F	For 3.2" display. Cut from larger strip
1	-	Header 4x1M	Install on the display; they usually have a 14 pin male header pre-installed but their 4 pin header is optional. Cut from larger strip
1	J8	2-pin R/A JST-XH socket	Battery socket soldered on PCB
1	-	2-pin JST-XH plug	Battery plug on lead to match J8
1	J9 (Optional)	Header 2x1M	Cut from larger strip
1	J10 (Optional)	Header 10x1M	Cut from larger strip
2	-	Header 20x1M	To mount Raspberry Pi Pico. Cut from larger strip
1	-	Header 5x1M	To mount PAM8302A module. Cut from larger strip
8	-	Header 1x1M	To mount PAM8302A + TP4056 modules. Cut from larger strip
1	L1	Blue 3mm LED	
1	R1	2K2	¼ watt. *Note: if the blue charge-indicator LED (L1) is too bright then replace this with a higher resistance value.*
1	R2	68R	¼ watt
1	R3	2R2	½ watt - **HALF WATT**
2	R4, R5	330R	¼ watt

2	R6, R7	27K	¼ watt
1	SW1	DPDT slide switch	AliExpress/eBay search: "6 pin slide switch dpdt"
6	SW8,SW4,SW2 SW5,SW6,SW7	12mm push button	AliExpress/eBay search: "push button switch 12x12x7.3mm"
6	-	-	Switch caps for the above
3	SW3, SW9, SW10	6mm push button	AliExpress/eBay search: "4-pin push button switch 6x6x5mm"
2	TP1,TP2	-	GND test points, optionally connect with half-loop of wire
1	U1	Raspberry Pi Pico	Programmed with the PicoMite firmware, see Chapter 3.
1	U2	PAM8302A - Mono 2.5W Class D Audio Amplifier	AliExpress/eBay search: "pam8302a amplifier module"
1	U3	TP4056 - Lithium Battery Charger	AliExpress/eBay search: "tp4056 lithium battery charger module". **Ideally replace R3 with 2k2; see Appendix D.**

1	VR1	20K Dual-Trimmer	This is a Linear (B-type) potentiometer. AliExpress/eBay search: "double gear tuning potentiometer B203"
1	TFT display with SD card reader	3.2" or 2.8" 320x240 pixel ILI9341 with touch	AliExpress/eBay search: "3.2 inch tft ili3941"
1	LiPo battery	3v7, 1000+ mAh	Sizes 404050, 523450, 503450 are suitable - avoid anything beginning with a 10, they are too "fat" (10mm). **Alternatively** a Nokia BL-5C battery can be used - carefully solder leads to the GND/VBAT+ terminals, leaving the middle "BSI" terminal unconnected. AliExpress/eBay search: "3v7 lithium polymer battery 1000mAh".
1	Speaker	8R, 2W	AliExpress/eBay search: "8 ohm 2 watt laptop speaker square"
8	Threaded screws	M3, 5mm long	Screws and spacers are only required at the corners of the PCB.
4	Tapped plastic or nylon spacers	M3, 10-12mm long	

Appendix C - Alternative firmware and software installation instructions

Chapter 3 describes the simplest way of installing the PicoMite firmware and GameMite software, however this has the disadvantage of replacing/erasing the entire contents of the Raspberry Pi Pico's flash memory.

For those who like to live dangerously, this appendix describes an alternative more manual process[2].

Loading the firmware

The Raspberry Pi Pico that underlies the GameMite comes with its own built-in firmware loader that is easy to use. Just follow these steps:

1. Follow the links from https://github.com/thwill1000/mmbasic-gamemite#firmware-and-software to download the latest compatible "PicoMite<version>.uf2.zip".

2. Extract the "PicoMite<version>.uf2" file.

3. Using a USB cable, plug the Raspberry Pi Pico into your computer (Windows, Linux or Mac) while holding down the white BOOTSEL button on the Raspberry Pi Pico (you will need to remove the TFT display to do this).

4. The Raspberry Pi Pico should connect to your computer and create a virtual drive (the same as if you had plugged in a USB memory stick) called "RPI-RP2". This drive will contain two files which you can ignore.

5. Copy the "PicoMite<version>.uf2" file to this virtual drive.

6. When the copy has completed, the Raspberry Pi Pico will restart and create a virtual serial port on your computer. The LED on the Raspberry Pi Pico will blink slowly indicating that the firmware with MMBasic is now running.

   If the LED does not start blinking, then it is possible that the Raspberry Pi Pico's flash memory was in a previously corrupted state. To fix this install the file https://github.com/thwill1000/mmbasic-gamemite/raw/main/download/flash_nuke.uf2 following the same instructions. This should reset the Raspberry Pi Pico to its factory fresh state from where you can try installing "PicoMite<version>.uf2" again.

Whilst the virtual drive created by the Raspberry Pi Pico looks like a USB memory stick it is not; the firmware file will vanish once copied and if you try copying any other type of file it will be ignored.

Loading the firmware will erase the flash memory including the current program, any programs saved in flash memory slots and all saved variables. So **make sure that you backup this data if you ever upgrade the firmware.**

The next step of the installation requires connecting a desktop computer to the Raspberry Pi Pico using a USB cable and then running a terminal emulator program on the desktop computer.

---

[2] In reality these are the original installation instructions before it was discovered it was possible to just use a single .uf2 file. They are maintained because they may prove useful if and when there is upgraded PicoMite firmware or GameMite software to install.

Virtual serial port

The virtual serial port created by the firmware acts like a normal serial port but it operates over USB.

Windows: Windows 10 + 11 include a driver for this virtual serial port. Once connected you should note the COM port assigned by the computer to the new connection. This can be done by starting Device Manager and checking the "Ports (COM & LPT)" entry for a new COM port as shown on the right.

Linux: A modern Linux distribution should present the Raspberry Pi Pico as an ACM device, e.g. /dev/ttyACM0 without requiring any additional driver. *Note: you will need to be a member of the "dialout" group to access it.*
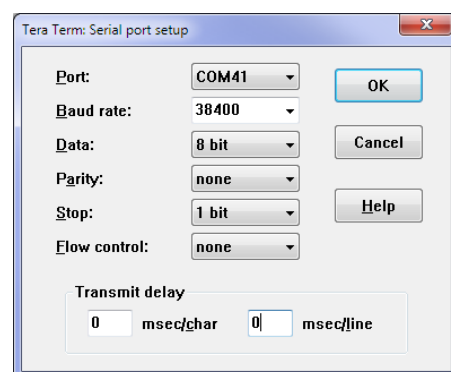
Terminal emulator

You will need a terminal emulator program on your desktop computer. This program acts like an old fashioned computer terminal where it will display text received from a remote computer and any key presses will be sent to the remote computer over the serial link. The terminal emulator that you use should support VT100 emulation as that is what the editor built into the GameMite expects.

Windows:

Tera Term (https://tera-term.en.lo4d.com) is recommended for Windows users as this has a good VT100 emulator and is known to work with the XModem protocol which you can use to transfer programs to and from the GameMite.

The screen shot on the right shows the setup for Tera Term:

- The "Port:" setting will vary depending on which USB port your Raspberry Pi Pico was plugged into.
- The GameMite ignores the baud rate setting so it can be set to any speed (other than 1200 baud which puts the Raspberry Pi Pico into firmware upgrade mode).

Linux:

The standard `screen` command can be used to connect, e.g.

```
screen /dev/ttyACM0
```

The console

Once you have identified the virtual serial port and have connected your terminal emulator to it you should be able to press return on your keyboard and see the MMBasic prompt, which is the greater than symbol (e.g. ">").

This is the console and is the most fundamental way to interact with the standard PicoMite. For the GameMite we just require it to configure the PicoMite firmware and install the GameMite software suite (menu, tools and games.)

If you wish to find out more then please see the PicoMite User Manual:
https://geoffg.net/Downloads/picomite/PicoMite_User_Manual.pdf.

Configuring the PicoMite firmware

The PicoMite firmware should be configured by typing the following OPTIONs at the console; some of them will cause the GameMite to reboot and depending on your terminal emulator you may need to reconnect:

```
OPTION PLATFORM "Game*Mite"
OPTION SYSTEM SPI GP6,GP3,GP4
OPTION CPUSPEED 252000
OPTION LCDPANEL ILI9341,RLANDSCAPE,GP2,GP1,GP0
OPTION TOUCH GP5,GP7
OPTION SDCARD GP22
OPTION AUDIO GP20,GP21
OPTION MODBUFF ENABLE 192
```

Installing the GameMite software

Although it is possible to run the GameMite software directly from the SD card (B:/) it is recommended that it is copied onto the GameMite internal flash drive (A:/) as described here:

1. Follow the links from https://github.com/thwill1000/mmbasic-gamemite#firmware-and-software to download the latest "GameMite-<version>-alternative.zip".

2. Open the "GameMite-<version>-alternative.zip" file and copy/extract the "GameMite" directory to the root of a FAT32 formatted SD card.

    Once extracted the SD card should contain the following:

    ```
    /
        GameMite/
            install-a.bas
            menu.bas
            startup.bas
            ...
    ```

    N*ote: the SD card may have other pre-existing content, but only the "GameMite/" directory is relevant to these installation instructions.*

3. Connect to the GameMite from a terminal emulator as described above.

4. Insert the SD card into the GameMite.

5. Run the following command at the console to copy the software to the A:/ drive:

    ```
    RUN "B:/GameMite/install-a.bas"
    ```

6. When this has completed RUN the following commands to complete setup:

    ```
    FLASH ERASE 1
    LOAD "A:/GameMite/startup.bas"
    FLASH SAVE 1
    OPTION AUTORUN 1,NORESET
    ```

    Note: FLASH ERASE 1 will report a harmless error if flash slot 1 is already empty.

7. The GameMite software should now be set up. If RESET is pressed or the device is power-cycled it should reboot to show the GameMite splash-screen before displaying the menu.

Appendix D - Miscellanea

A collection of gritty details that you will hopefully never need to know.

WARNING! Before making any potentially damaging modifications to the ILI9341 TFT module please consider posting a question on "The Back Shed" forum: https://www.thebackshed.com/forum/Microcontrollers.

Audio circuit

The GameMite uses pins GP20 and GP21 for the PWM sound output. These outputs are each fed to low pass filter circuits comprising R4/C9 and R5/C10 which convert the PWM signals to audio. The roll off frequency of these filters begins at approximately 4800Hz. Capacitors C12 and C13 AC couple the audio signal to resistors R6 and R7 which are used to set the approximate maximum gain. Increasing their value will decrease the maximum gain and vice versa.

Resistors R6 and R7 both connect to the volume control pot (VR1) and so will combine the two sound outputs into one mono signal. The output of the volume control is fed to the digital amplifier module (PAM8302A) and from there to the speaker. The PAM8302A operates at around 250KHz and the output is not further filtered beyond what is present on the module, but fed straight to the speaker as is.

The components R3, C3 and C5 decouple the power supply for the PAM8302A to reduce noise generated by the PicoMite's power supply.

Adjusting the trimpot on the PAM8302A module

In addition to the main volume control pot (VR1) the PAM8302A module has a miniature 10K trimpot on it which can be adjusted to set the maximum volume using a screwdriver with a small flat blade.

DO NOT use a Phillips/cross-head screwdriver; despite appearances to the contrary the screw is not of that type and you will possibly chew up the head.

This is easiest done after the software has been installed by following this procedure:

1. With the power off, carefully remove the TFT display module.
2. Attach the GameMite to your desktop computer via USB.
3. Switch the power on.
4. Connect to the GameMite via a terminal emulator as described in Appendix C.
5. If the GameMite menu is not shown in the terminal emulator[3]:
    a. Press Ctrl-C
    b. The prompt ">" should be displayed
    c. Type RUN
6. Navigate to the "Sound Test" program.
7. Navigate to "Play Music".
8. Play "Spring (Vivaldi)".
9. Set the GameMite thumbwheel pot to maximum volume - fully anti-clockwise.
10. Adjust the miniature trimpot on the PAM8302A to the desired maximum volume.

---

[3] Depending on the terminal font / code page the characters displayed for the menu border may be wrong, it doesn't matter.

Powering the GameMite

The GameMite's LiPo battery is charged by connecting the Raspberry Pi Pico to an active USB power source with SW1 in the OFF/CHG position. Charging and overcharge protection are managed by the TP4056 module powered from the Raspberry Pi Pico's VBus pin.

When purchased a TP4056 module is typically fitted with a charge current programming resistor R3 with a value of 1k2, this sets the maximum charging current to 1000 mA which is suitable for LiPo batteries with a capacity of 1000 mAh or greater. For safety reasons **DO NOT** use a battery with a capacity of less than 1000 mAh unless you also change R3 appropriately. See the TP4056 datasheet (http://www.tp4056.com/d/tp4056.pdf) for a table of recommended values for R3 to suit alternative battery capacities.

In light of the above it is recommended that:

- The GameMite is never left to charge unsupervised.
- A 3v7 LiPo battery with a **minimum** capacity of 1000 mAh is used.
- If you feel comfortable doing so, the programming resistor R3 on the TP4056 module can be changed to 2k2 to reduce the charge current from 1000 mA to ~500 mA. This will provide a greater safety margin for the charging current at the expense of a longer charging time.

The charging behaviour can also be modified by fitting the "Charge mode" jumper (J9):

- It is **recommended** that it is not fitted in which case the LiPo will only charge when the GameMite is OFF, i.e. SW1 is in the up (OFF/CHG) position.
- If it is fitted then the LiPo will charge irrespective of whether the GameMite is ON or OFF. Note that with the link fitted if SW1 is in the ON position then the GameMite will be powered from the USB, not the battery, i.e. even in this state the battery will not be driving a load at the same time it is charging.

The 3mm blue charging-indicator LED (L1) will be lit if USB power is being applied to the TP4056 module (from the Raspberry Pi Pico's VBus pin). *Note: this LED doesn't show the actual state of the charge and will remain illuminated even if the battery is fully charged*.

In addition the TP4056 module has two onboard surface mount LEDs, one blue (or green) and the other red. The red LED will be illuminated whilst charging and the blue once full charge has been reached. If the red LED is flashing then the battery is faulty, or not attached. *Note: these LEDs will be partially hidden under the TFT.*

Note: the USB connector on the TP4056 module can be ignored, it shouldn't be possible to connect it because of other components in the way, and if you did succeed it should not represent a problem.

SD card troubleshooting

The SD card reader on the ILI9341 TFT modules includes 1K resistors in series on its data lines. Shorting out these resistors (or replacing them with 0R) has been suggested as a fix for SD card reliability problems. In practice we have not seen this issue with the GameMite and in any case the quality of the SD card is probably more important. See picture overleaf.

TFT display brightness

On some, older, ILI9341 TFT modules the brightness of the display can be controlled by decreasing (brighter) or increasing (dimmer) the value of R2 (default 68R). More recent modules supply the power to the backlight via a transistor (Q1 on the module) and for these displays R2 does not control the brightness. See picture overleaf.

TFT 3v3 regulator

The ILI9341 TFT module has an onboard 3v3 regulator (U1 on the module) so that they can be used from a 5V supply. The GameMite only supplies 3v3 so after the regulator dropout the module will be working at < 3v3. In practice this has not proved to be a problem (the ILI9341 will work at 2v8 and possibly below). If power to the TFT seems to be a problem then the regulator can be shorted out with a solder blob between the pins of J1 on the module. See picture overleaf.

WARNING! After shorting out the regulator the TFT module should not be connected to a 5V supply for fear of releasing the "magic smoke".

TFT SPI screen layout

Q1 used to control backlight on modern displays,

1k Resistors in series with the SD card lines.

3v3 Regulator, U1, and bypass jumper J1.