PicoTamachibi

A Tamagotchi like toy based on Raspberry PI Pico

Initial idea:

PICOTAMACHIBI

Picotamachibi

12 June 2023
10 minute read

By Kevin McAleer
Share this article on

VIDEOS

Watch the associated videos here:

WHAT IS PICOTAMACHIBI?

`Picotamachibi` is the name for a fun MicroPython based virtual pet.

Difficulty: Intermediate
Type: project
Categories:

pico micropython pets games

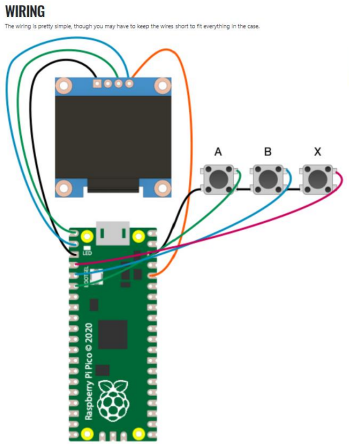
Kevin McAleer

I build robots, bring them to life with code, and have a whole load of fun along the way

- [Homepage]
- [GitHub]

BILL OF MATERIALS

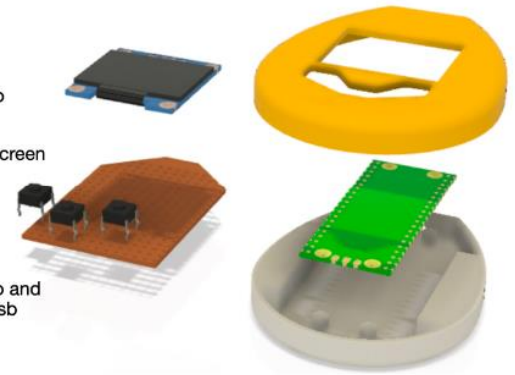
Item	Description	Qty
Pico	Raspberry Pi Pico / Pico W (either will do)	1
3x Buttons	3x Tact switches	3
Wire	Red, Black, Green, Blue wire & solder	1
OLED Display	SSD1306 128x64 OLED Display	1
Veroboard	Small strip of veroboard	1



3d print case

Fusion 360

- 3d printed case in two parts
- Top with cutouts for screen and buttons
- Veroboard for the tact switches and ssd1306 screen
- Bottom holds the pico and has a cutout for the usb connector



Building the project:

Ziele:

1. Ein Tamagotchi mit RPI Pico, Display und Knöpfen bauen
2. Arbeiten mit Code-Objekten
3. Strukturiertes Vorgehen beim Erschließen von Softwareprojekten
4. Ein bestehendes Projekt erweitern

Details:

- Zusammenbau der Hardware:
  - RPI auf Breadboard einsetzen
  - Buttons einsetzen
  - Komponenten Verbinden
- RPI Pico zum laufen bringen
  - Micropython installieren (UF2-File)
  - Thony DIE auf eurem PC installieren
  - Tamachibi-Projekt übertragen
  - Debugging bis es läuft
- Den Code erschließen und Verstehen
  - Fertigt ein UML-Modell des Codes an
- Den Code und das Projekt erweitern (Beispiele):
  - Hardware-erweiterungen (mehr Buttons?, Sound?, LEDs?)
  - Scrolling Toolbar [< A B (C) D E >][< B C (D) E A >]
  - Game als Tamagotchi Funktion [Happy +1]
  - Ein richtiges Spiel implementieren
  - Das Tamagotchi steuern
  - Save States (Tamagotch vergisst alles beim Neustart)
  - Tamagotchi „Entwicklungen“
  - Turbo-Mode
  - Inverted (Dark-Mode)
  - Skin (umschaltbares design)
  - Lowrers Mode
  - Multiplayer / Network /Communication

Bonus:

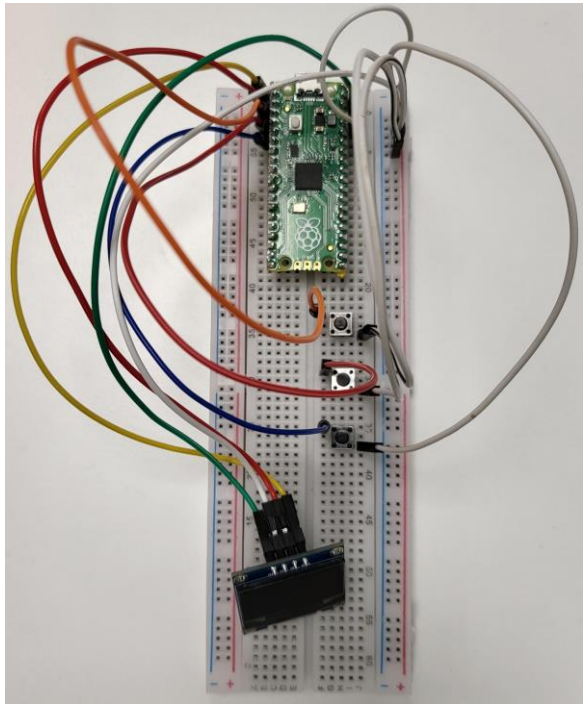
- Wer möchte darf sein Tamagotchi mit einer Hülle versehen und behalten
- Es gibt ein MASTER-TAMAGOTCH das alles kann

Begrenzungen:

- Arbeit in ~2er Gruppen
- Jede Gruppe baut ein Tamgotchi
- Dokumentiert eure Codeerweiterungen und stellt sie der großen gruppe vor

Zeitvorgaben:

Fertigstellung bis Freitag, 02.02.2024 EOD (15:00)

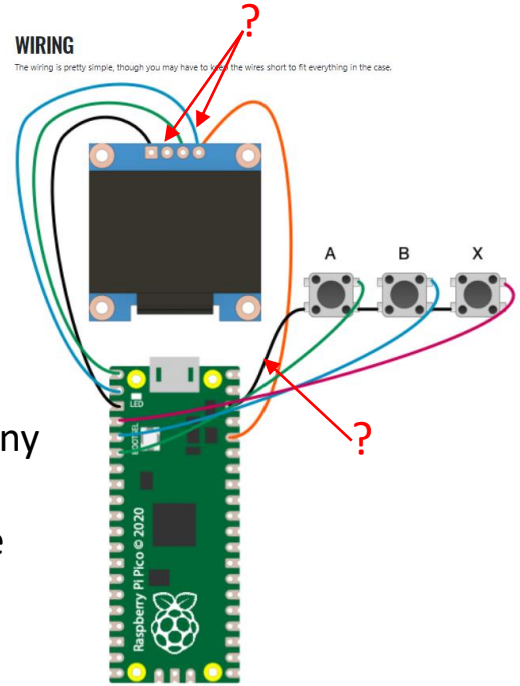


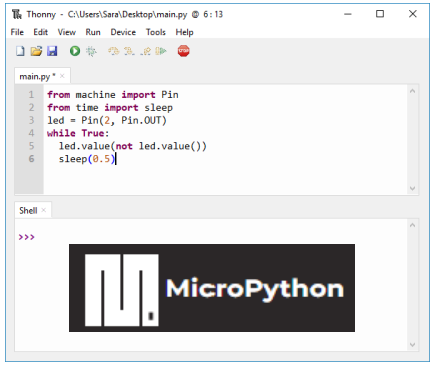
- Task sheet (german)

Picotamachibi.pdf

Lessons learned

- There are problems with the suggested **wiring**
- More explanation on how to flash the RPI with the correct **firmware** is needed
- Information on how to configure **Interpreter** and **COM port** is needed in Thonny
- A guide on how to **view files** and transfer the source code to the RPI is missing
- The code is buggy and inefficient

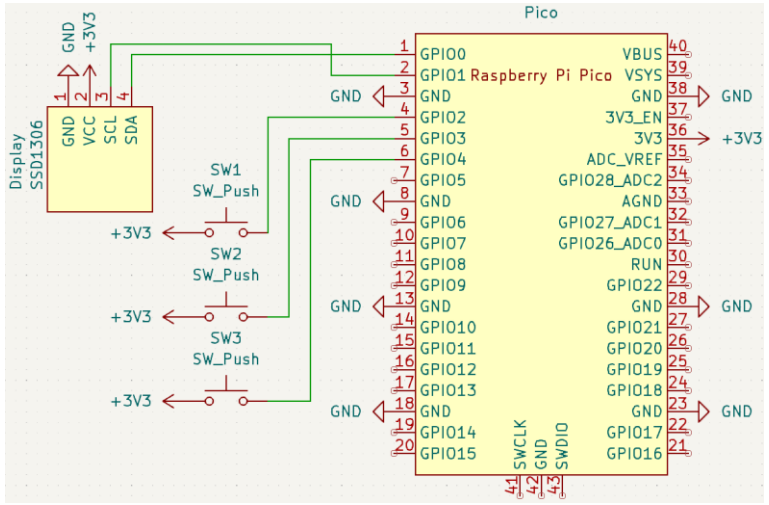




- .uf2 file (latest)
- Thonny

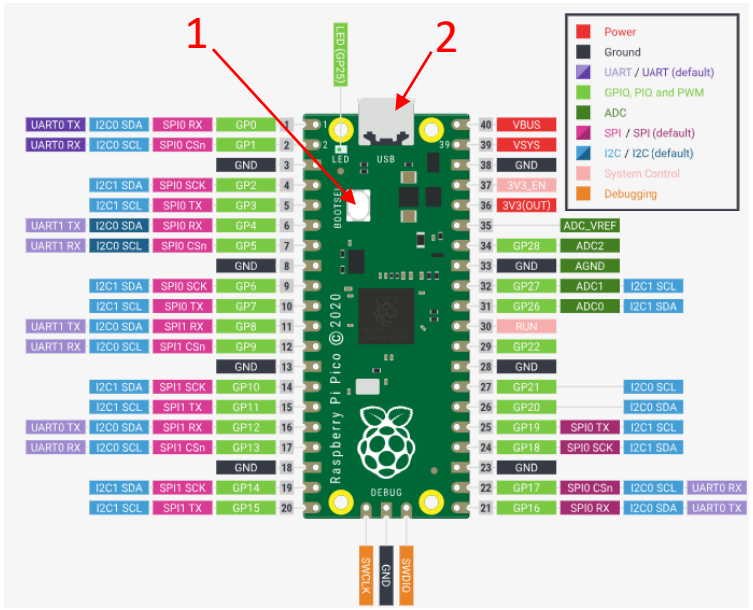
Filling in the gaps:

Fixing the schematic



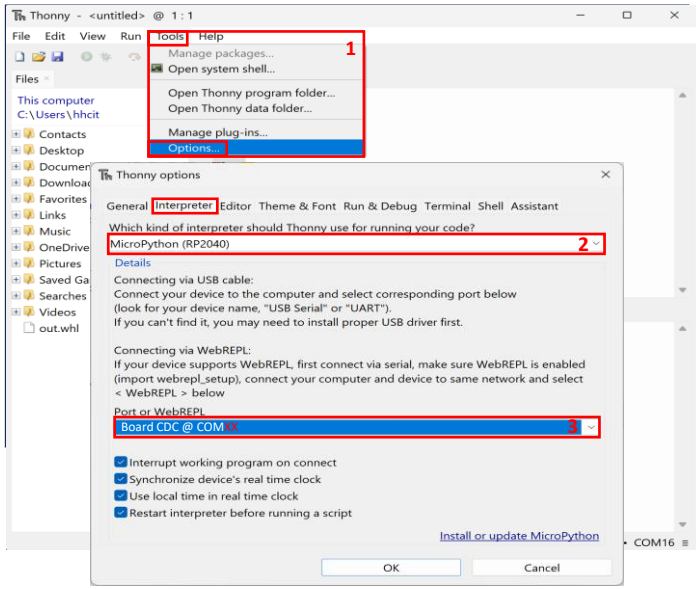
- **Input Pins** for buttons can be connected to **HIGH (3.3V) or LOW (GND)** (reference)
- The source code has to match how the buttons are wired **(PULL_UP vs. PULL_DOWN)**
- The Tamachibi code expects the pins to be connected to HIGH (3.3V)
- **TIP:** Connect 3.3V and GND to the **“rails” (red & blue)** on one side of the breadboard and **connect** the **buttons and peripherals** to them

Flashing the firmware (.uf2)



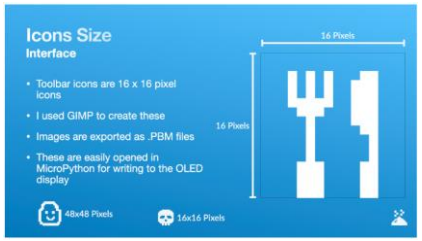
- **Hold** the **button** on the RPI Pico
- **Connect** the **USB cable** to the RPI and the Computer
  - The RPI appears as device in your **file manager**
- **Copy the .uf2** to the RPI
  - The RPI disconnects automatically *(pling sound)*
- **Disconnect** the USB cable
- **Reconnect** the USB cable

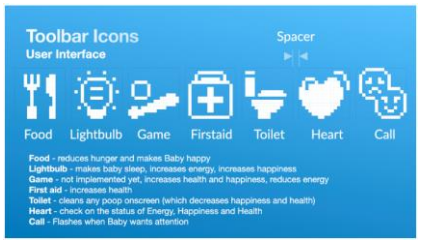
Selecting Interpreter & COM port

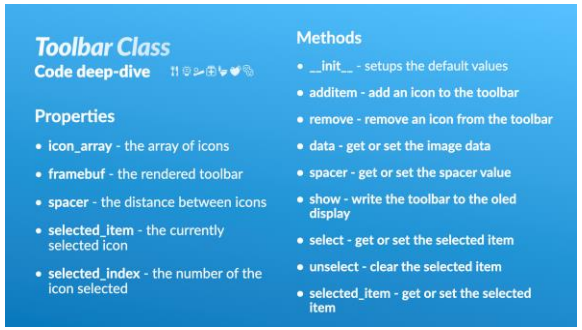


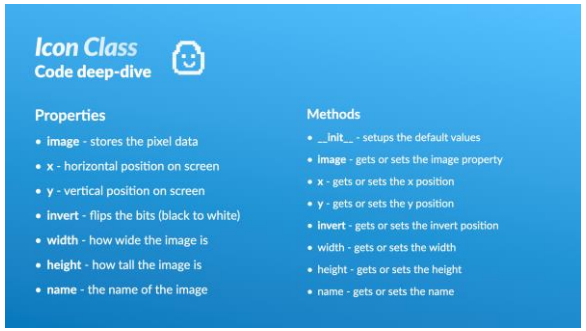
- Select **Options** from the **Tools** menu
  - Select the **Interpreter** tab
- Select **Micro Python (RP2040)** as Interpreter
- Select the **COM port** of your device
  - The number varies e.g. COM14
  - You have to repeat the port selection each time you connect a new RPI

The source code:



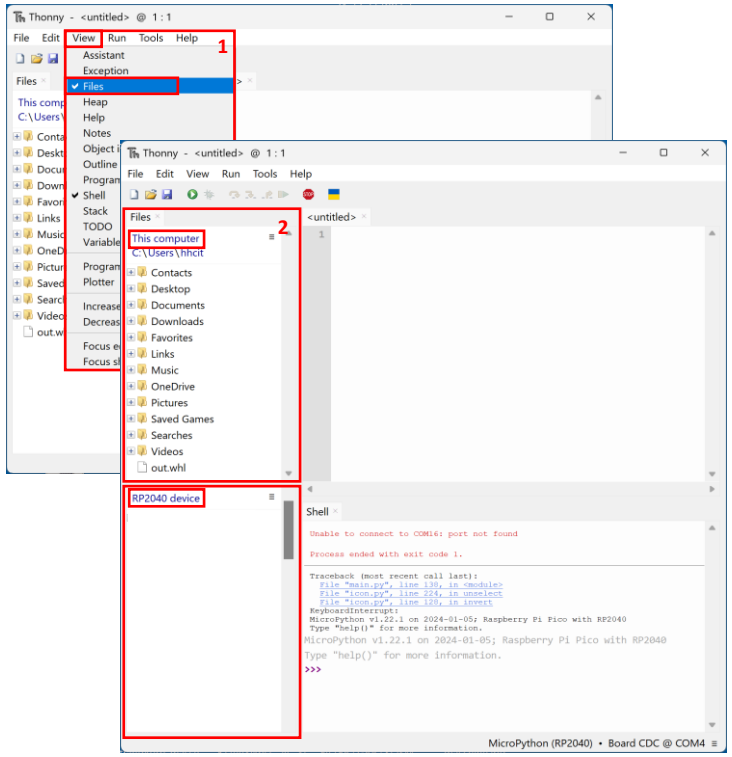






- There are also .py files for button testing and other peripherals
- Use the test scripts to simply verify if the respective peripheral is working as expected

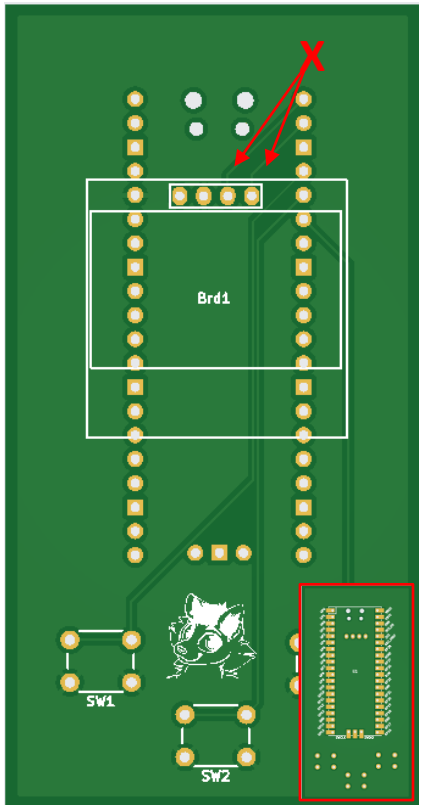
Copying the files

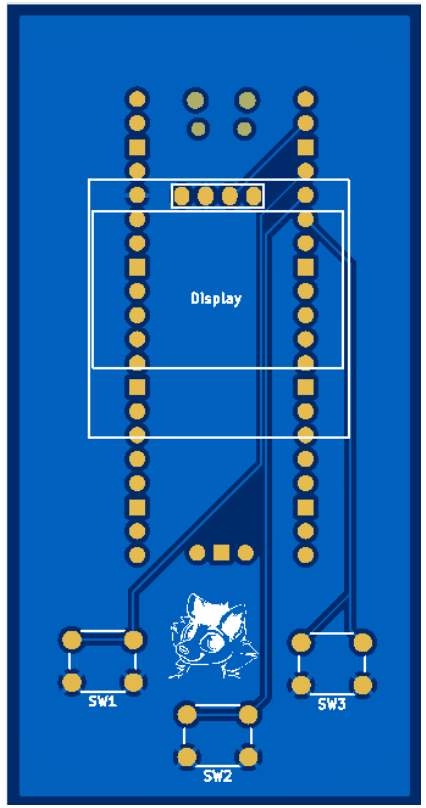


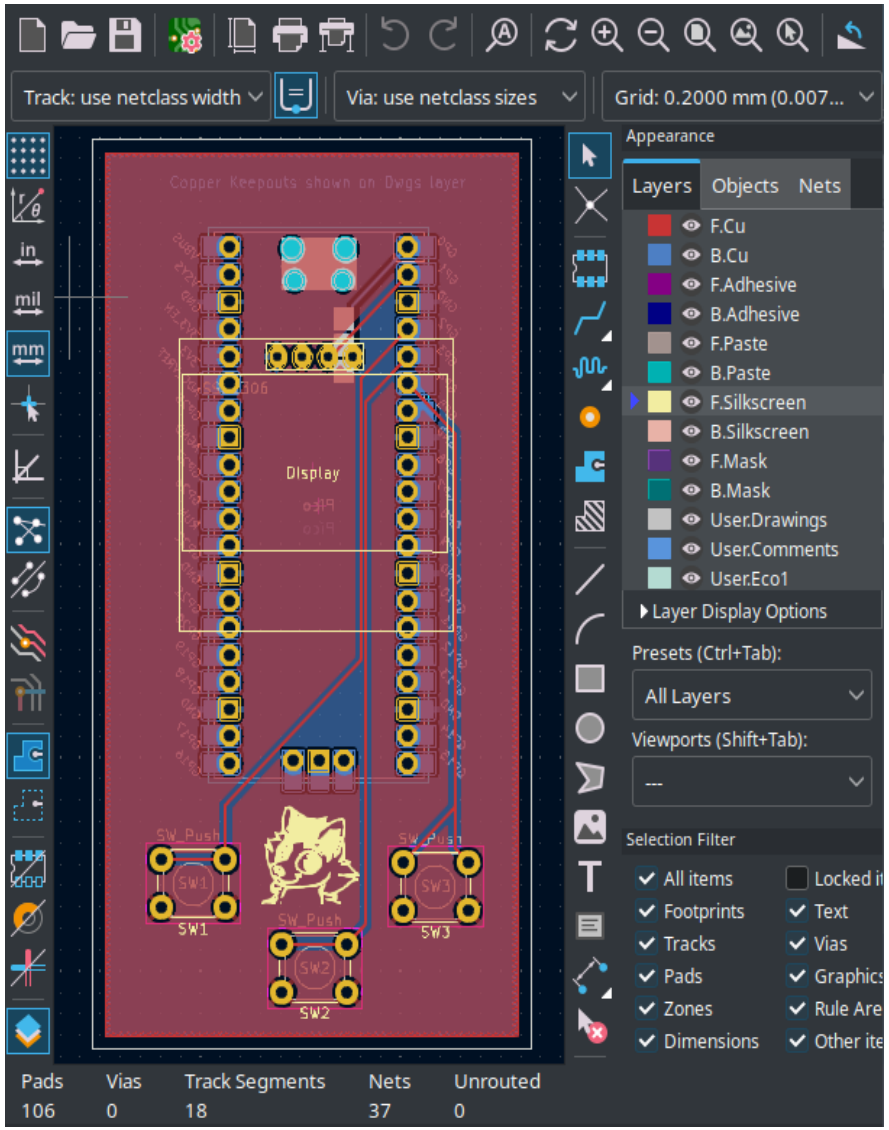
- Activate **Files** in the **View** menu
- **Copy** the project to the RPI
  - All **.PY** and **.pbm** files
  - UI in the file selection is *fiddly*
  - Space on the RPI is limited you can not simply copy the whole repository

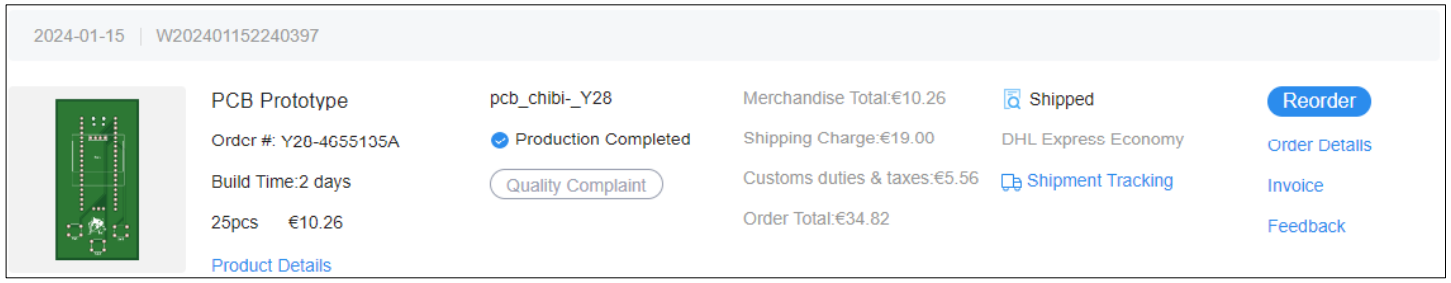
Level UP:

- KiCad
  - A design tool for PCBs
  - & schematics
- JLCPCB
  - Fab ordering
  - Pricing
- About V1.2
  - Ups
  - V1 fix

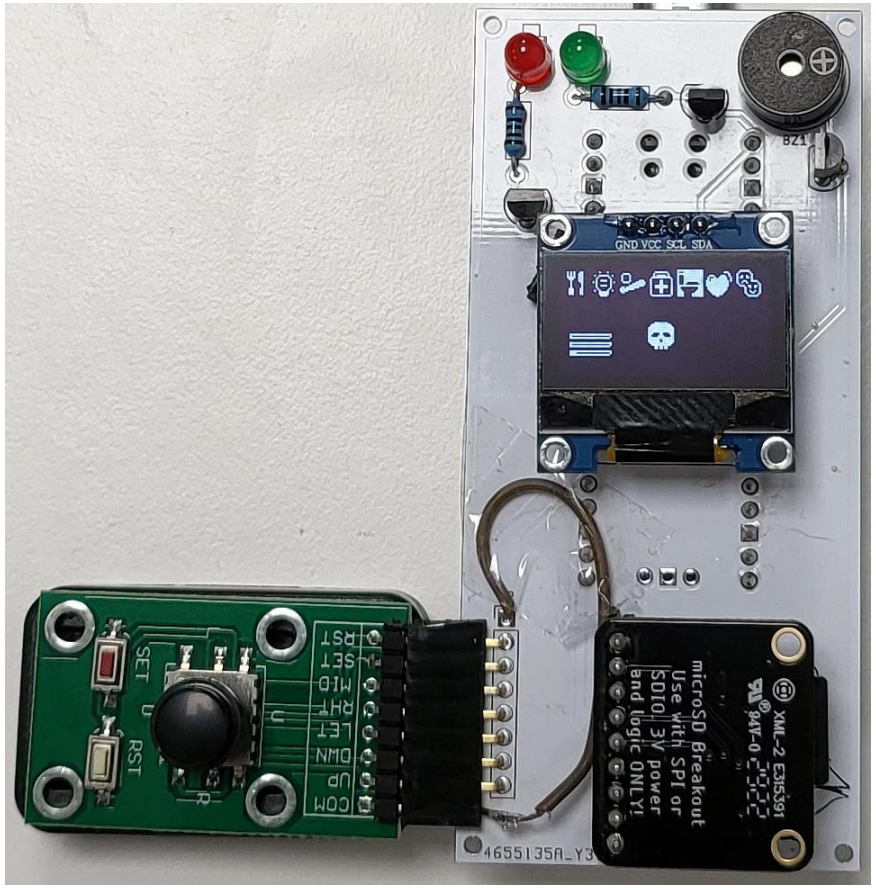


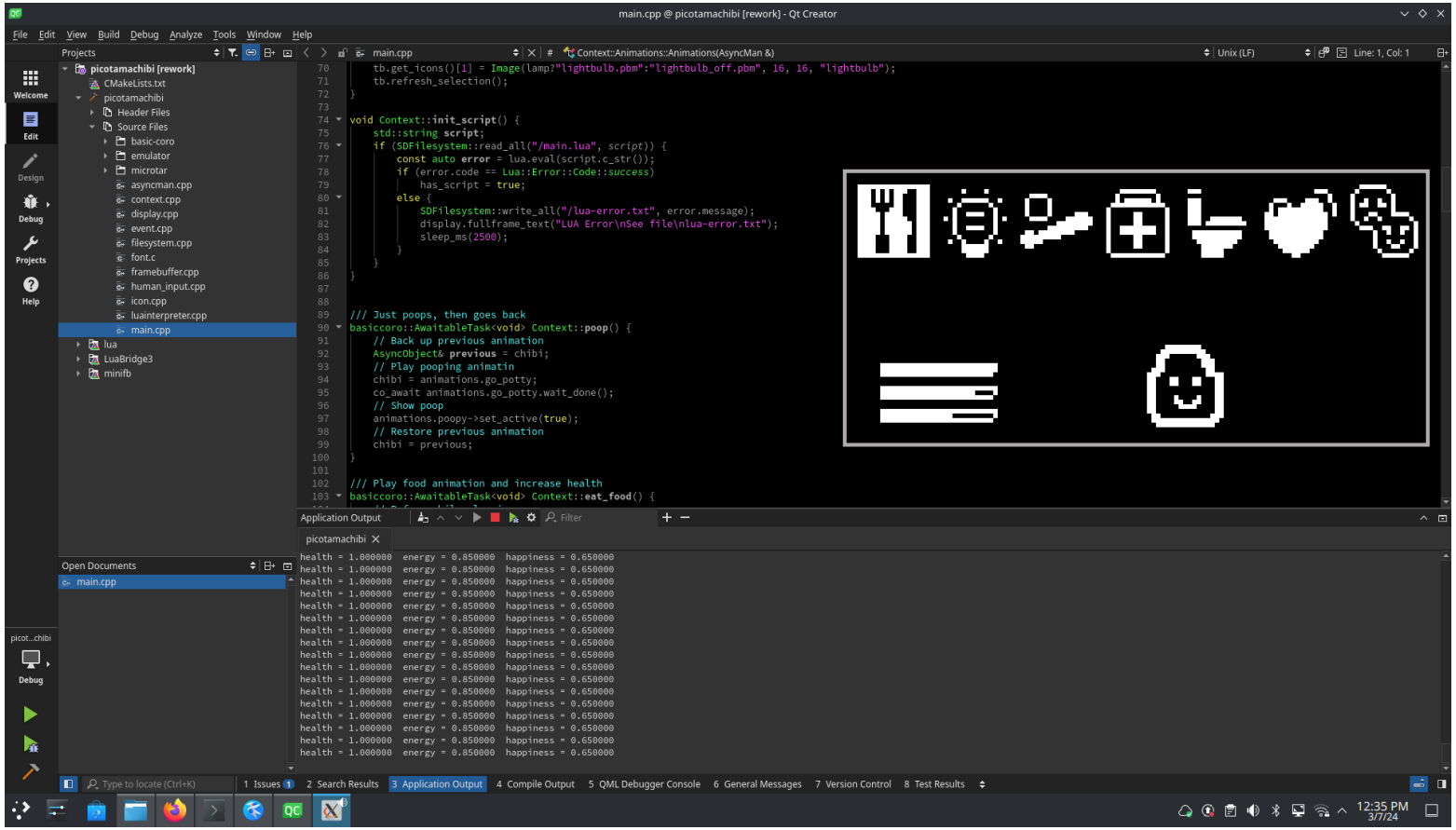






The C++ version:





Features

- +90% Performance boost (runs on V1 & V2)
- More standard compliant code
- Bugfixes and improvements
- Displays energy levels
- More complex character behavior
- Desktop emulator

Applied mods:

Hardware

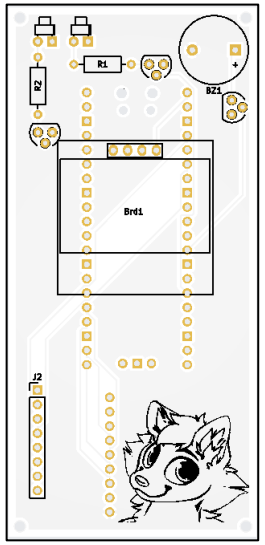
- Board development and iterations
- Joystick
- SD card reader
- Buzzer (passive)
- LEDs (red & green)
- Power input pins
- Upgrade to RPI Pico W (on some units)

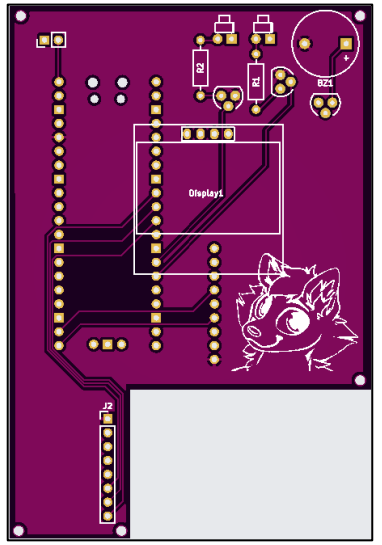
Software

- Menu ribbon
- Dynamic button controller
- Save state on SD card

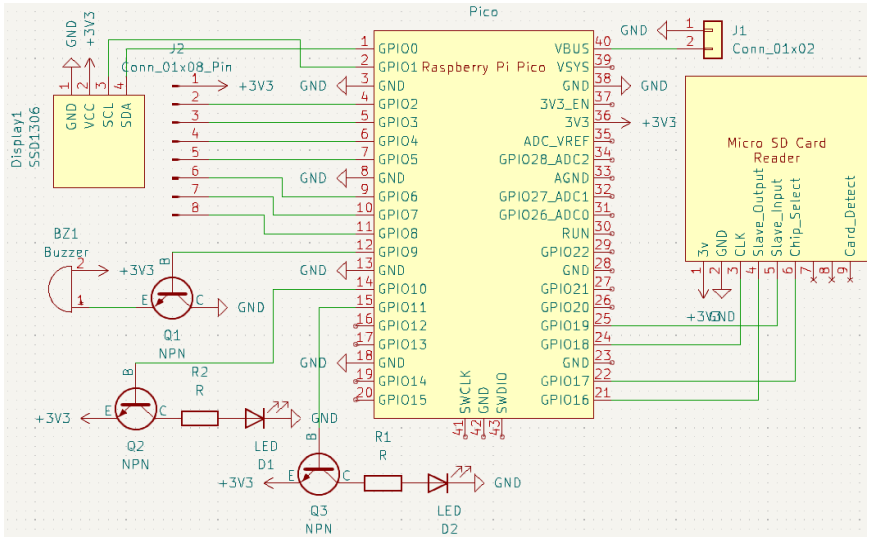
ToDo

- Charging circuit and lipo
- Breakout pins
- More coding
- Multiplayer
- …



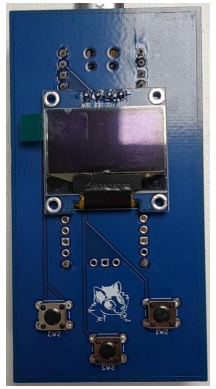


Picture



Final builds:

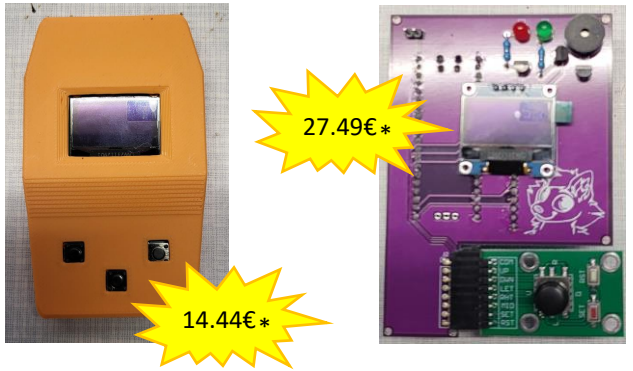
- Pictures



Takeaways & future:

- Github
  - Our improved code
  - The C++ version
  - V1 & V2 KiCad files
  - V1 & V2 gerber files
  - Documentation (this)
  - Basically everything!
- Bill of materials
- Calculation
- The box

Item	Qty	Cost € (all)	Cost € (1 build)	Comment
SSD1306 128 x 64 I2C OLED Display	5	22.00	4.40	Link (Amazon)
Buttons	100	5.00	0.16	Link (Amazon)
PCBs	25	37.00	1.48	From JLCPCB, includes shipping and taxes. The PCBs only V1 11.64€ / V2 16.43€
Pin header (40 pins wide)	30	7	0.40	Link (Amazon) Technically only 10 pins per unit are needed, the calculation assumes a nice build with all pins soldered
RPI Pico	1	8	8	Link (Amazon) Prices vary greatly. You can get them cheaper. Beware of scammers and fakes ☺
SD card reader	1	9	9	Link (Amazon) 3.3V is important. Those from Adafruit are great but hard to get and expansive
Joysticks	5	8	1.60	Link (Amazon)
Passive Buzzer	20	7	0.35	Link (Amazon) 12mm x 8.5mm, PASSIVE
LEDs, resistors, transistors	x	x	2	Links (Amazon) transistors, resistors, LEDs calculated generously



*Case not included

Thank you:

Contributors

- Kevin McAleer (initial idea)
- BBWHH IT22 group and friends, especially:
  - Nils (Mastermind, C++ version, KiCad)
  - Daiman (KiCad & Prototyping)
  - Julian (coding)
  - Justin (coding & VsCode integration)
  - TheFlow & Luca (cases)
  - Alex (project planning, documentation, coding, soldering, cash cow & whip)
- Hardware manufacturers and driver producers (SDcard & Display)
- JLCPCB & KiCad
- The Internet

Q&A

Backstage:

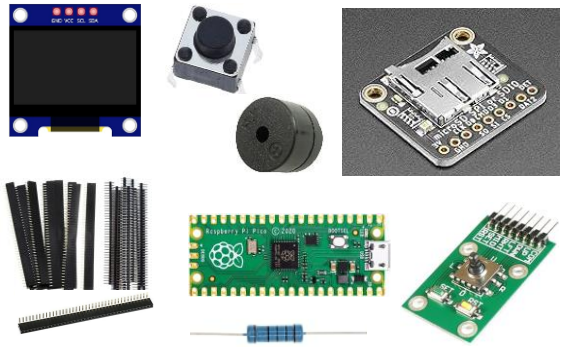
Additional thoughts:

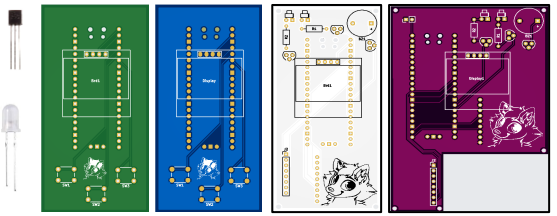
- What about a better and possibly colored display?
  - Cool
  - Generally possible
  - Pricing
  - Performance in Micro Python poor, better in C++
  - Requires more GPIO pins
  - Difficult trade-off
- Can it run DOOM?
  - There is a project (RP2040 Doom)
    - Github
    - Blog post
  - We did not try it
    - Probably not on the exact Tamachibi hardware
    - Would be a separate project
- Can it emulate a Gameboy?
  - Same as with Doom
  - Github
- Can it run Crysis?
  - Probably not :)

What is in the box?

(or should be)

- Parts (basic Tamachibi)
  - Displays
  - Buttons
  - Pin headers M&F (& angled)
- RPI Picos
- PCBs
  - V1.0 (green)
  - V1.1 (blue)
  - V2.0 (white)
  - V2.1 (purple)
- Documentation
  - Presentation (this)
  - Worksheet (pdf)
- Additional parts (V2)
  - NPN-transistors
  - 100r resistors
  - LEDs
  - Passive buzzer
  - Joysticks
  - SD card reader
- Online content
  - Github (this project)





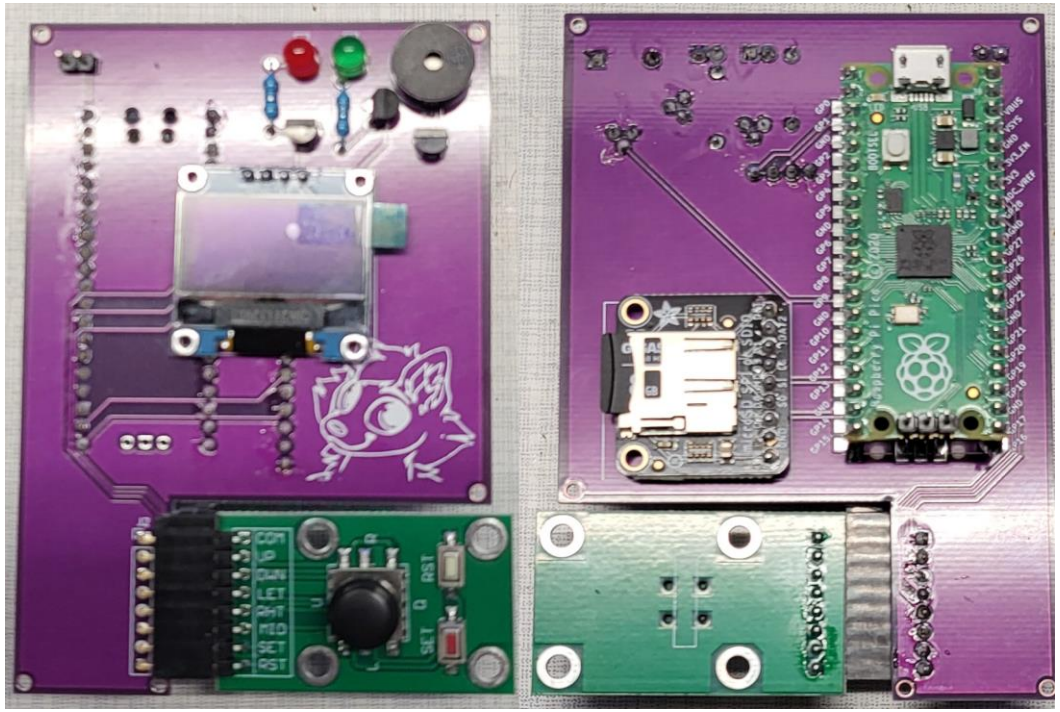
Bonus content:

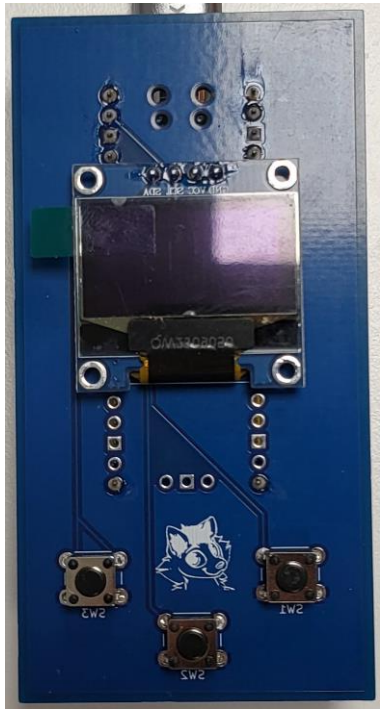
- Presoldered V1 PCBs
  (not perfect but working with a *twist*)

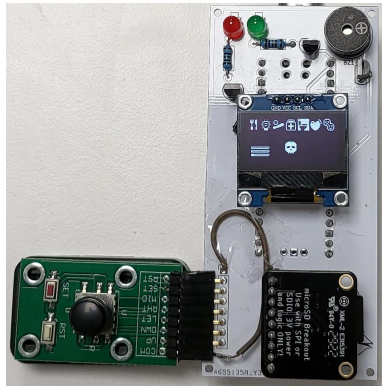
Not in the box:

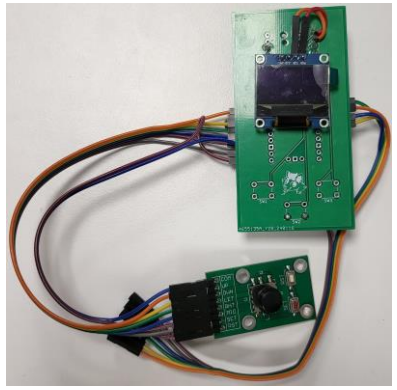
- Micro USB cable
- Bread board
- Dupond / jumper cable
  M->M & M->F
- Everything related to soldering

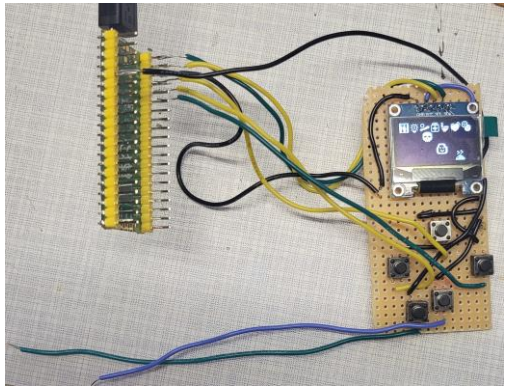
Gallery:







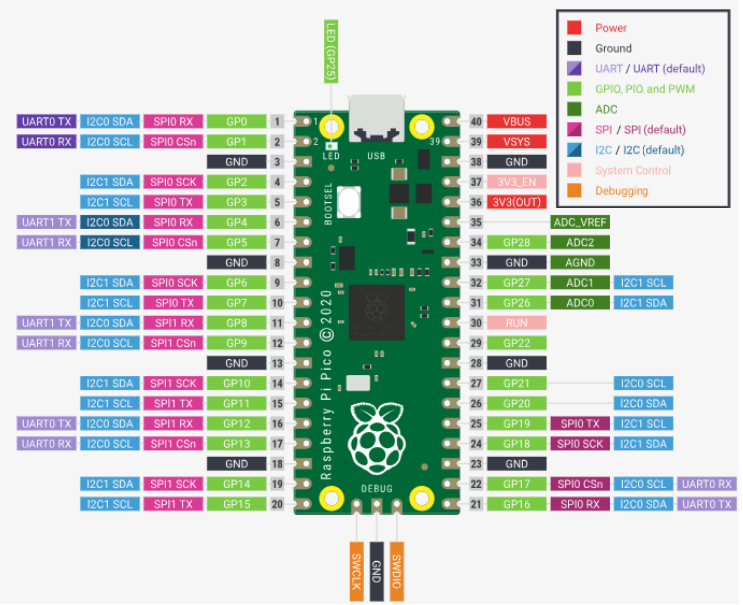




Aftershow:

Headline

- Topic
- Other topic (on some units)

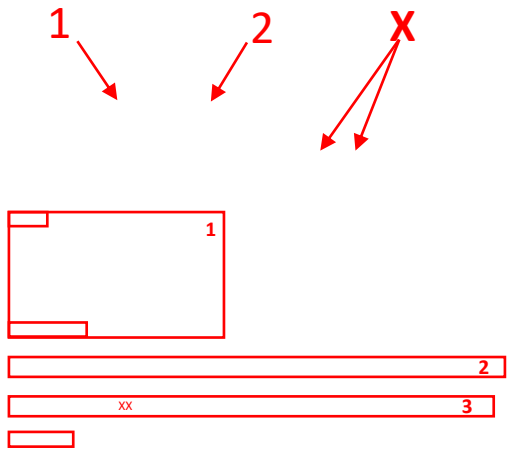


- Topic
- GitHub

- Arbeitsblatt

Picotamachibi.pdf

Placeholder



Good News:

- Spread some love

Attention:

- Bla bla bla