Read before you start

This repository contains some information and ideas for building a nixie calculator and is provided *as-is* for educational, experimental, and hobbyist purposes only. However, this repository is only intended for suitable qualified electronics engineers. The project has not been tested for compliance with electrical, safety, or electromagnetic standards (e.g., CE, FCC).

WARNING: Nixie tubes and associated circuitry operate at potentially dangerous voltages. If you choose to build or use the calculator, you are doing so at your own risk.

Please read all documents in the Docs directory before deciding whether to build the calculator.

About the arbitrary-precision arithmetic

The arbitrary-precision arithmetic used in the calculator is experimental.

It's based on a slightly modified version of the ratpak library. I had to make some changes to compile the library for an ESP32. In addition, I have tried to fix some memory leaks, rounding problems, and faulty formatting routines.

Calculations with very big numbers can be slow. For example, calculating the square root of 9e+9999 with a precision of 32 digits takes about 8 seconds. For calculations in the more common range between 1e+99 and 1e-99, the result is usually displayed immediately. One exception is the calculation of non-integer factorials where a gamma function approximation is used.

The precision of the calculations and the internal registers has been set to 32 as a compromise between accuracy and performance.

The decision to use arbitrary-precision arithmetic was determined by the programming fun and by the poor precision of the 64-bit floating-point arithmetic. Just an example: if you calculate 1.0000000000001 ^ 999 using the C/C++ pow function you will get 1.00000000009982 instead of 1.00000000009990.

Precision of timer, stopwatch and clock

The precision of the timer and stopwatch depends on the internal MC oscillator. It is possible to connect the (probably) more precise 32kHz clock signal of the RTC chip to the ESP32, but I have no plans to implement this change in the firmware. If no GPS module is connected, the clock depends on the accuracy of the RTC chip.

Nixies

To extend the lifespan of the nixie tubes, it's recommended to use the auto-off function or to connect the peripherals module for presence detection (PIR). It's also recommended to use the cathode poisoning prevention function. This is especially important for the IN-15A symbol nixies which only display + and -.

Firmware development tools

I use VS Code and the ESP-IDF extension with arduino-esp32 as a component to develop the firmware for the controller (ESP32). For the keyboard firmware (ATmega328P) I use the Arduino IDE.