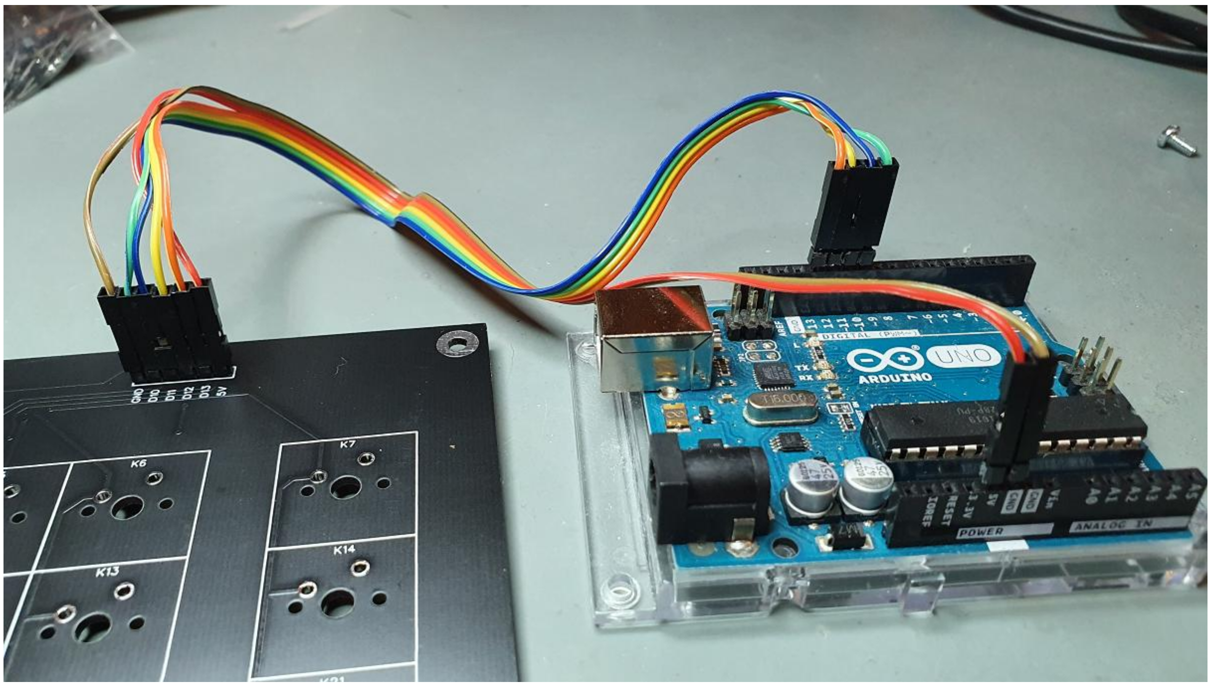
Flashing the bootloader and the firmware

Note: the pictures are only intended for a better understanding of the instructions and may not correspond to the latest hardware revision.

Burning the keyboard bootloader

The keyboard uses an ATmega328p-AU. The bootloader must be burned before flashing the firmware. With an Arduino UNO and the Arduino IDE, the bootloader can be burned directly on the keyboard. I recommend doing this before soldering the switches.

Remove all connections from the keyboard. Then connect the BOOTLOADER header pins to the Arduino UNO (see silkscreen).



Connect the Arduino UNO to your PC and start the Arduino IDE.

Select the Arduino UNO board:

Tools->Board

Select the com port:

Tools->Port->(your com port)

Open the ArduinoISP example:

File->Examples->11ArduinoISP->ArduinoISP

Upload the sketch to the Arduino UNO:

Sketch->Upload

Select the programmer:

Tools->Programmer->Arduino as ISP

Flash the bootloader:

Tools->Burn Bootloader

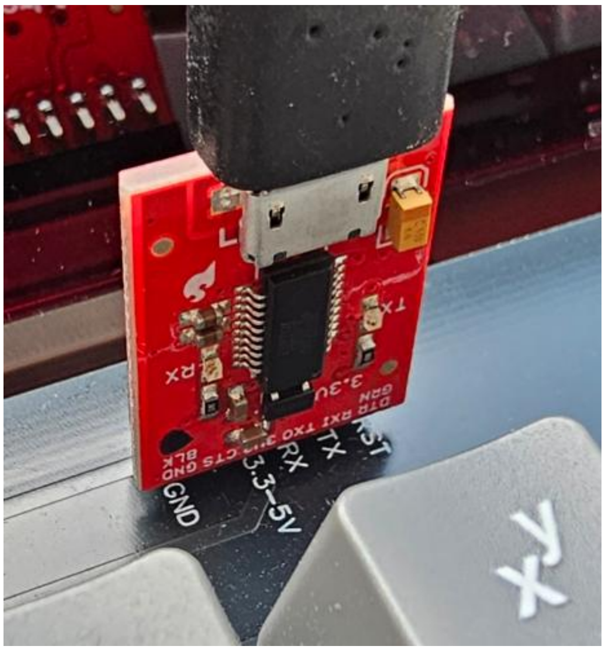
Check output for errors

Flashing the keyboard firmware

For the keyboard I use either the Arduino IDE or Visual Studio Code with the PlatformIO extension.

(see Software/Firmware/Keyboard/README_HowToBuild.md).

Power off the calculator and connect a **3.3V** FTDI adapter to the FTDI header on the keyboard. Make sure you connect your adapter the correct way around by checking the ground pin (GND). Then compile and flash the firmware. Check output for errors.

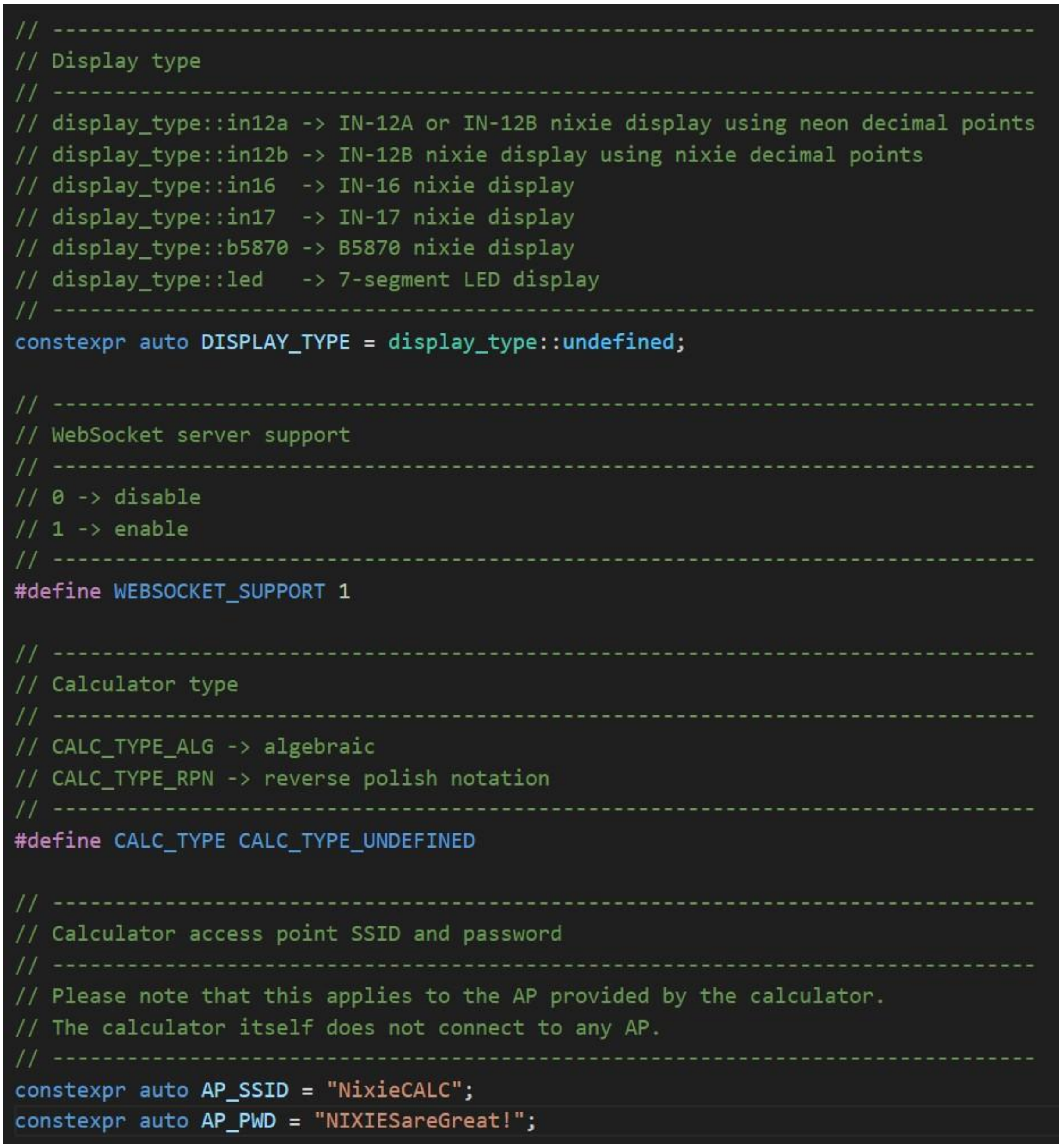


Flashing the controller firmware

For the controller, I use Visual Studio Code with the ESP-IDF extension.

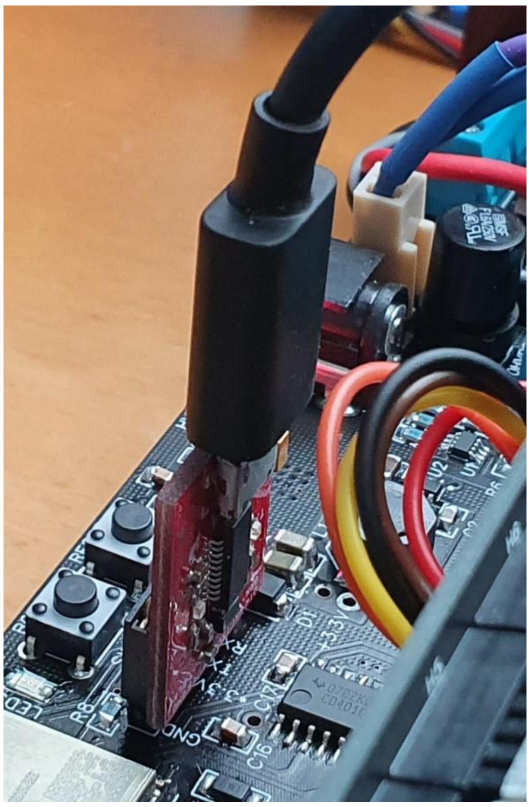
(see Software/Firmware/Controller/README_HowToBuild.md)

Before compiling, you must adapt the include/config.h file to your requirements. Define your display type, enable or disable WebSocket support and select your preferred calculator engine.

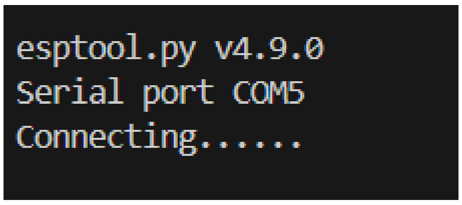


```
// -------------------------------------------------------------------------------
// Display type
// -------------------------------------------------------------------------------
// display_type::in12a -> IN-12A or IN-12B nixie display using neon decimal points
// display_type::in12b -> IN-12B nixie display using nixie decimal points
// display_type::in16  -> IN-16 nixie display
// display_type::in17  -> IN-17 nixie display
// display_type::b5870 -> B5870 nixie display
// display_type::led   -> 7-segment LED display
// -------------------------------------------------------------------------------
constexpr auto DISPLAY_TYPE = display_type::undefined;

// -------------------------------------------------------------------------------
// WebSocket server support
// -------------------------------------------------------------------------------
// 0 -> disable
// 1 -> enable
// -------------------------------------------------------------------------------
#define WEBSOCKET_SUPPORT 1

// -------------------------------------------------------------------------------
// Calculator type
// -------------------------------------------------------------------------------
// CALC_TYPE_ALG -> algebraic
// CALC_TYPE_RPN -> reverse polish notation
// -------------------------------------------------------------------------------
#define CALC_TYPE CALC_TYPE_UNDEFINED

// -------------------------------------------------------------------------------
// Calculator access point SSID and password
// -------------------------------------------------------------------------------
// Please note that this applies to the AP provided by the calculator.
// The calculator itself does not connect to any AP.
// -------------------------------------------------------------------------------
constexpr auto AP_SSID = "NixieCALC";
constexpr auto AP_PWD = "NIXIESareGreat!";
```

Power off the calculator and connect a 3.3V FTDI adapter to the controller board. Make sure you connect your adapter the correct way around by checking the ground pin (GND).



Build the firmware and start flashing the firmware. Once you get the "Connecting…" message, press and hold the "boot" button on the controller board, then press and release the "reset" button and finally release the "boot" button. This should start the flash process. Check output for errors.



```
esptool.py v4.9.0
Serial port COM5
Connecting......
```

Check firmware versions

After flashing the controller and the keyboard firmware, **remove the FTDI adapter** and power on the calculator. During start-up, the calculator briefly shows the controller firmware version on the left and the keyboard firmware version on the right.