Edge-Lit NTP Clock

Firmware Manual

Version 4.44

Mitch Markin

June 21, 2024

A clock with NixieCron “M” displays running this firmware,
showing the Nixie tube simulation.

Table of Contents

Introduction

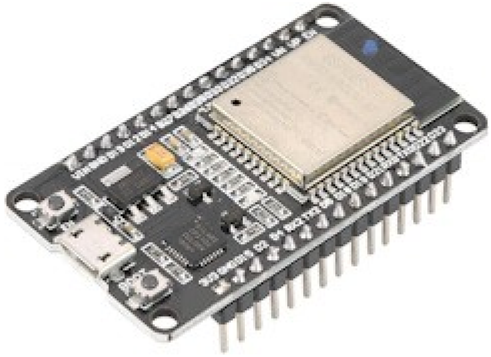
This firmware uses an ESP32 development board (devkit) to operate a digital clock using LED numeric displays that were inspired by Nixie tubes.

Like Nixie tubes, each display used for the clock has the digits 0 to 9 stacked one in front of another and the appropriate digit is lit to display it. However the digits are engraved on clear acrylic panels instead of being electrodes shaped like numerals inside a glass tube filled with neon. The panels are edge-lit from the bottom with WS2812B multi-color LEDs to illuminate the engravings. So instead of being high voltage devices that can only show amber digits like Nixie tubes, these displays run on 5 volts and the LEDs can show pretty much any color.

Several types of displays can be used. The **Hardware** section of this manual has more details about the displays and their availability.

ESP32 is a series of low-cost and low-power System-on-a-Chip (SoC) microcontrollers developed by Espressif Systems. They include integrated WiFi capability, dual-core processors, and much more.

Numerous manufacturers make ESP32 development boards that also include 4MB of flash memory, USB connectivity, a 3.3-volt power regulator, a couple of LEDs, and buttons for reset and programming. This is all packaged on a convenient little circuit board about a third the size of a business card.



This firmware syncs with a Network Time Protocol (NTP) server over WiFi to automatically set the clock's time and keep it accurate. The firmware can also automatically find the local time zone and it adjusts for Daylight Saving Time on its own, too.

Besides the time and date, the clock can also show the temperature.

The firmware is menu driven and has numerous settings to control the clock's operation and the display colors, including a Nixie tube simulation. The menu items and their settings are shown numerically on the edge-lit displays. They can also be shown with text on a small OLED display.

All setup is done from the menu except entering the credentials for the WiFi network that the clock will use for time syncs. The firmware gets the ESP32 to serve its own web pages for this that can be accessed over WiFi from a phone or computer.

Compiling the Code

This firmware was developed using the **Platform IO IDE** and that's the best way to edit, compile, and upload it to an ESP32. PlatformIO has professional features that make developing code easier, such as checking the syntax as you write the code and autocompletion of variable names and function calls. It will also remind you of what parameters a function needs when you type its name. This all helps to make sure you don't write anything that “does not compute”.

PlatformIO can automatically find and install all the libraries the code needs. It can also automatically do most of the setup required to compile the code and upload it to a microcontroller.

And PlatformIO is fast. It typically takes about 20 seconds to compile this code (maybe as much as two minutes the first time when it has to find the libraries).

The **Arduino IDE** can also be used. The new 2.3 version is pretty slick and it comes close to matching PlatformIO's functionality. Adding libraries is easier now compared to the older 1.8 version but you still have to install them manually. It has autocompletion that works almost as well as PlatformIO's but it doesn't check for syntax errors as you write the code.

However, using the Arduino IDE to edit and compile this code will be a frustrating experience. Eight libraries will have to be added to the IDE and you will have to make sure they are the right ones and the right versions. Also the Arduino IDE takes about two minutes to compile this code and as long as six minutes if it's the first time. That's a long time to wait especially when developing code and you aren't sure if what you wrote will do what you want. Or maybe you'll wait a few minutes just to find that you made a silly mistake somewhere.

Like the Arduino IDE, PlatformIO is free to download and use without restrictions. The learning curve for PlatformIO is a bit steeper than it is for the Arduino IDE, but the effort is well worth it. PlatformIO's basic operation can be grasped fairly quickly, though. After that it has a lot of advanced features to explore if you choose to do so.

If you are getting the code from its GitHub repository, it's already set up to use with PlatformIO. Just download the repository to a local directory, open the directory in PlatformIO, and all the files will be there.

If the code was supplied on a flash drive or other media, all the files PlatformIO needs for this project are in the **EdgeLit_NTP_Clock_v4.44_PIO** directory.

For those not familiar with PlatformIO there are instructions for installing it and using it to compile the code in the **PlatformIO IDE Guide** on Page 70 of this manual.

But if you choose to use the Arduino IDE and the code was supplied on a flash drive or other media, all the files needed are in the **EdgeLit_NTP_Clock_v4.44_Arduino** directory.

If you are getting the code from its GitHub repository there are instructions in the Using Arduino IDE file there to help you set things up and compile the code.

There is also an **Arduino IDE Guide** on Page 68 of this manual that lists the libraries the code needs and the board setup.

Hardware

Edge-Lit Displays

This firmware works with the following edge-lit numeric displays:

NixieCron made by Led-Genial (any type, S, M, or L)

EleksTube made by EleksMaker (R version only, not IPS)

Lixie made by Connor Nishijima (Lixie Labs) (original version)

Lixie II made by Connor Nishijima (Lixie Labs) (updated version)

Display type is set in hardware by the two-bit binary number read at GPIO 25 and GPIO 26 on the ESP board (most significant bit on 26).

The ESP pulls the unconnected pins high so only the pins that have to be low are connected to ground. As shown in the schematic diagrams:

For **NixieCron** displays both pins are left open (11)

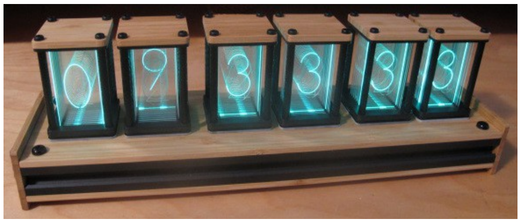
For **Lixie** displays GPIO 26 is left open and GPIO 25 is connected to ground (10)

For **EleksTube** displays GPIO 26 is connected to ground and GPIO 25 is left open (01)

For **Lixie II** displays both pins are connected to ground (00)

As of this writing NixieCron displays are available from Led-Genial in Germany.

EleksTube displays are available as a clock kit from EleksMaker in China. The microcontroller that comes with the kit is not an ESP32. It has no WiFi capability so it can't sync to NTP servers unless the clock is connected to a computer with Internet access. Therefore the kit's stock microcontroller board would have to be replaced with an ESP32 module to run this firmware.

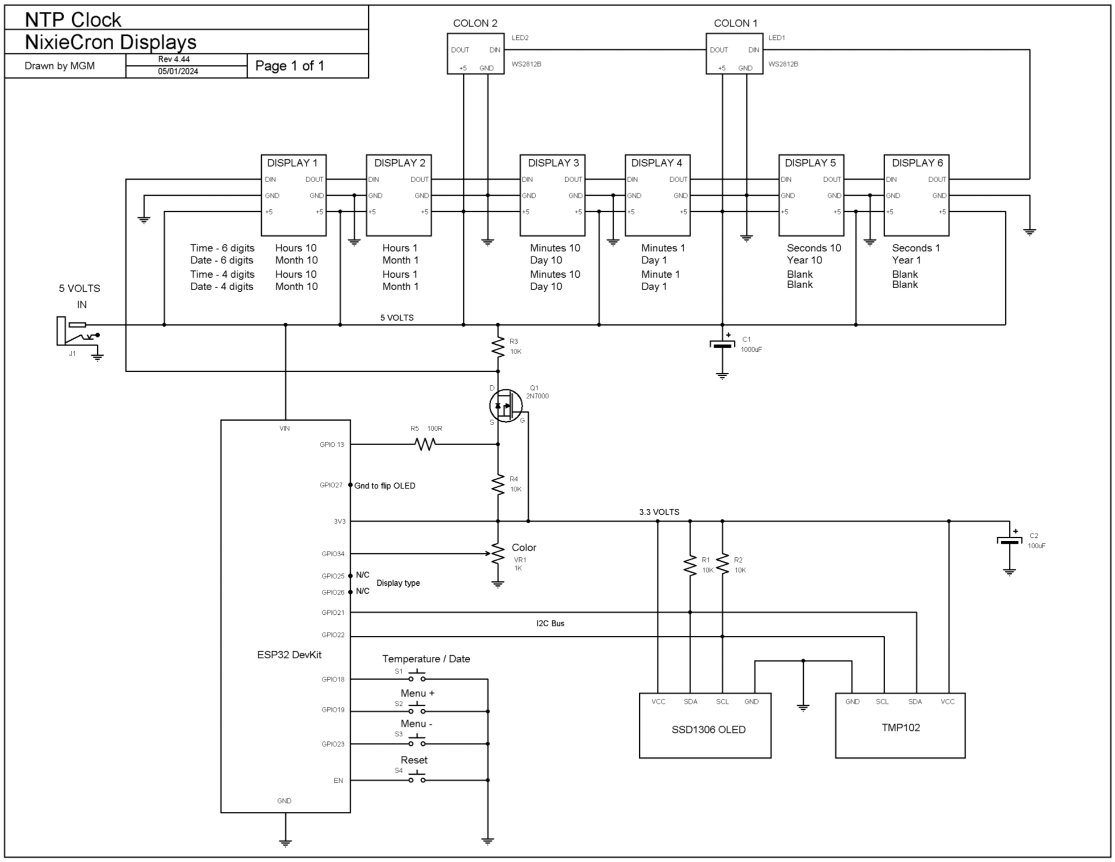


The original Lixie displays have been updated to Lixie II. The original version has been discontinued. The addressing scheme for the LEDs in Lixie II displays is different from the originals but this firmware should be able to work with them. However, this has not been tested because of their poor availability.

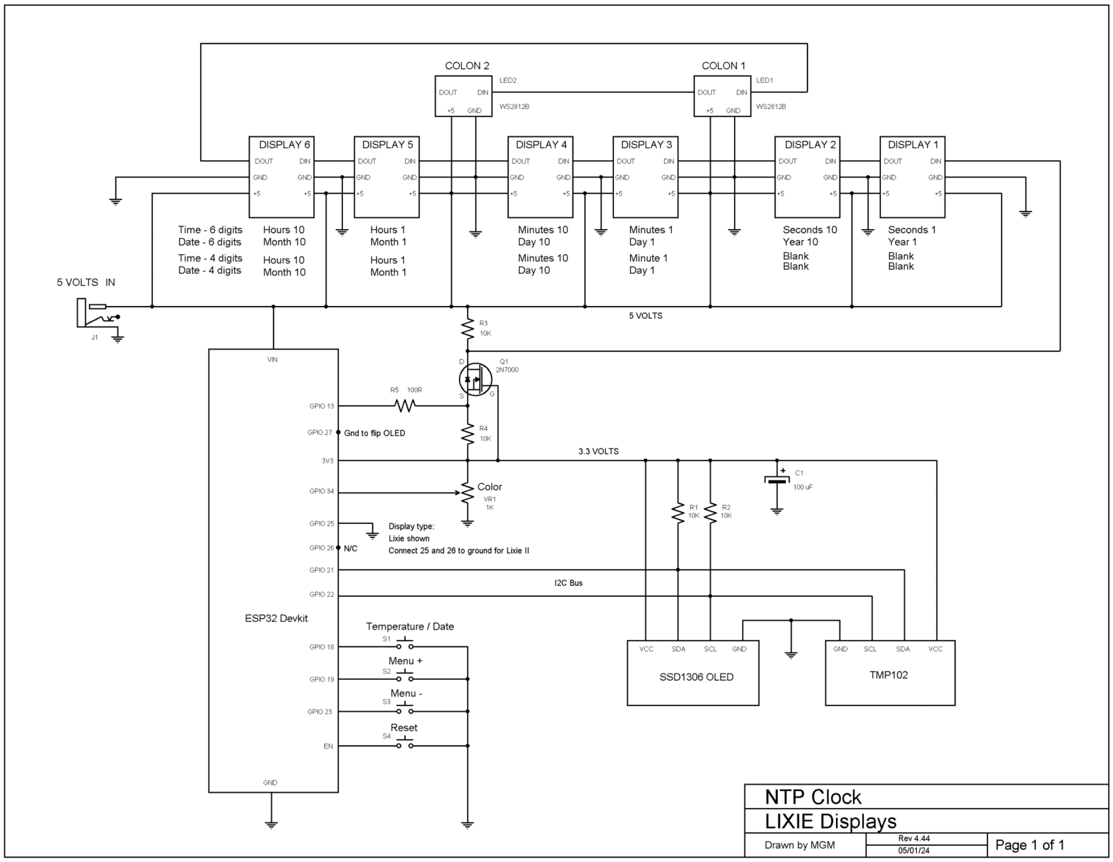
Lixie Labs still has a site on Tindie to sell Lixie II displays but it says they have been out of stock since 2020. If you are adventurous enough to make your own, though, the designs are open source and the CAM files to make the digit panels and circuit boards are available on Lixie Labs' Tindie site.

Schematic Diagrams

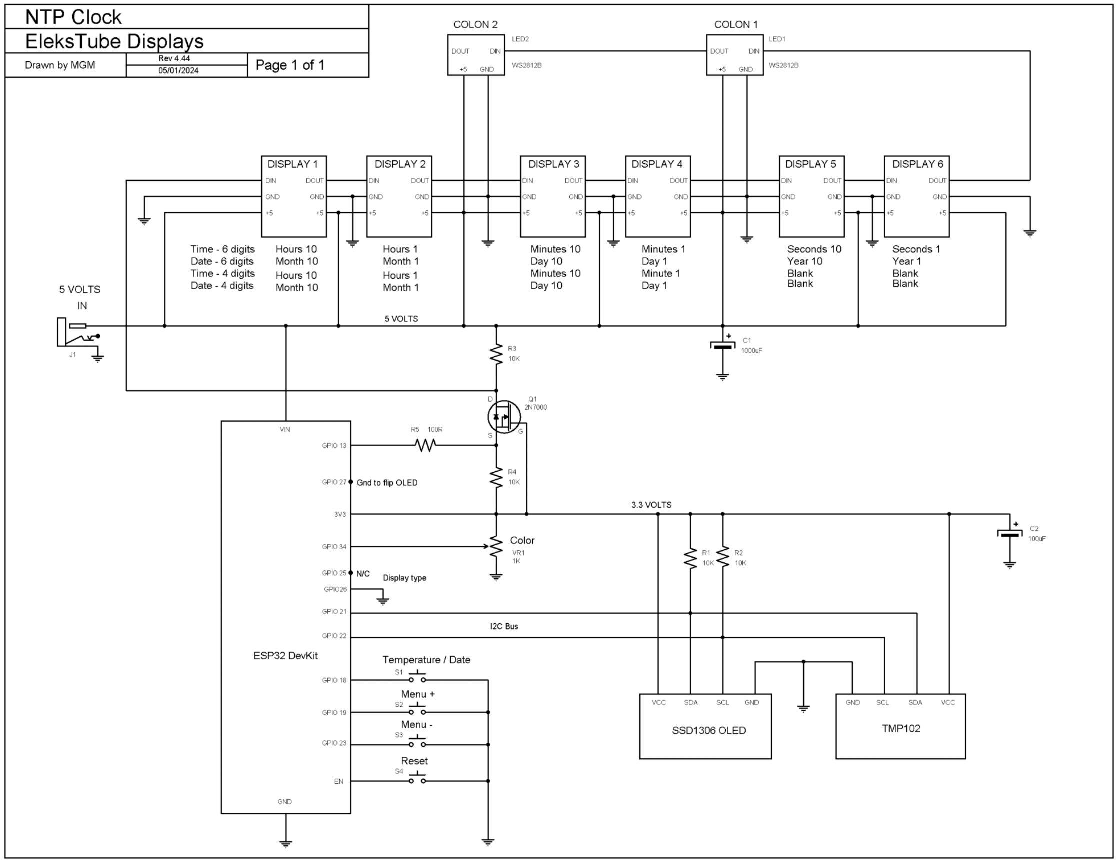
Schematic for NixieCron displays



Schematic for Lixie displays.



Schematic for EleksTube displays.



The main difference is Lixies get their data sent to the rightmost display. NixieCron and EleksTube data goes to the leftmost display. Also the connections to GPIO 25 and GPIO 26 on the ESP module are different because that's how the firmware knows what display type is being used.

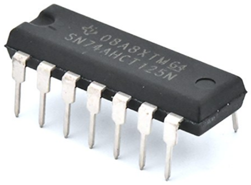
Note that +5 volts and ground is shown connected to each digit module, although the modules can simply be daisy chained together with power connected only at one end. However, the WS2812 string is 122 LEDs long. If power is only connected once the voltage drop by the time the last LEDs are reached can result in the colors and brightness not being quite right across the six displays. Therefore at the least, power should be connected at each end of the string.

Level Shifting

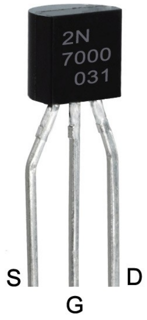
WS2812B LEDs expect 5-volt data for reliable operation. ESP microcontrollers are 3.3-volt devices therefore some kind of level shifting has to take place. A lot has been said about how to do this. Some say the LEDs work fine with no level conversion. Others say just use pull-up resistors and a diode or a transistor. Still others say it's best to use a dedicated IC or module to do the job and there are many to choose from. All of these solutions work, although some only work in very favorable conditions.

WS2812 LEDs are very picky about their data and they can act erratically if things aren't right. The data is an 800KHz stream with tight timing requirements. Also the rise and fall times of the square wave transitions have to be very fast. Each LED regenerates the data at 5-volt levels before passing it on to the the next one, so once the first LED gets good data everything should be fine.

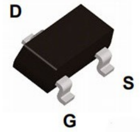
Using a dedicated IC such as a 74AHCT125 is generally said to be the best solution. It's a quad high speed bus buffer that can be run from the 5-volt power supply but it has no trouble reading 3.3-volt data at its inputs. Actually the chip's specifications say the data levels could be as low as 2 volts. However with these clocks this chip actually works too well. It picks up too much noise and the data it generates is useless.



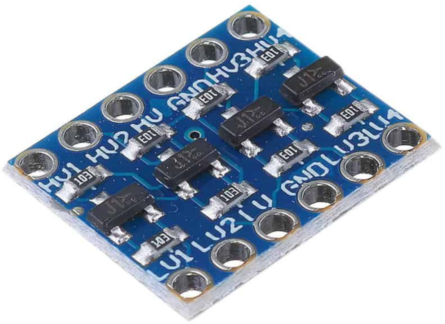
What has been found to work the best is using a MOSFET transistor and a couple of pull up resistors as shown in the schematic diagrams. The circuit can be built from scratch using a 2N7000 transistor and two 10K resistors. A BSS138 transistor may work better but it's only available as a surface-mount part. The circuit is also widely available as a four-channel module.



2N7000



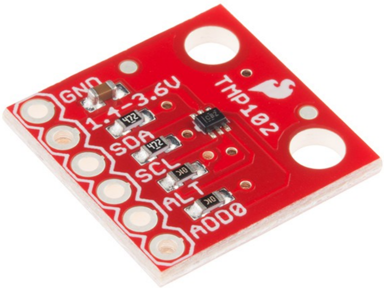
BSS138



Four-channel Level Shifter

Temperature Sensor

For the temperature function a SparkFun TMP102 module has to be installed. It's connected to the I2C bus on GPIO 21 and GPIO 22 on the ESP board.



It's a tiny board that costs less than the standard DHT22 sensors and it's more accurate.

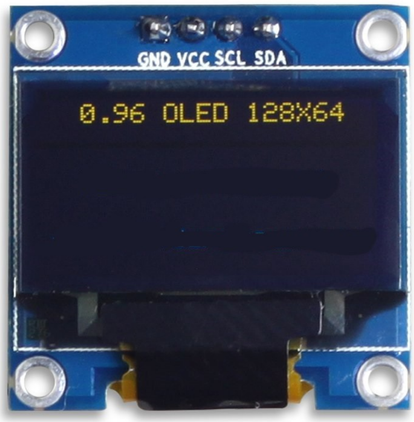
The TMP102 module doesn't have to be installed for this firmware to work. Temperature display functions are simply ignored if the sensor isn't present.

OLED Menu Display

An SSD1306 .96" OLED module can be installed to show the menu items and their settings. It connects to the I2C bus on GPIO 21 and GPIO 22 on the ESP board along with the temperature module (if used).

The OLED module is optional. This firmware works without it. The menu items are also shown on the edge-lit displays. However since the edge-lit displays can only show numbers they just show each menu item's number and another number that corresponds to the item's setting. Without the OLED display you won't have the text to say what the menu items or their settings do, unless you refer to this manual or you have a great memory!

Normally the display is mounted as shown in the picture with the four connection pins at the top. If it has to be mounted “upside down” with the pins at the bottom connect GPIO 27 to ground on the ESP board. This will flip the text so it appears properly. Otherwise just leave GPIO 27 unconnected.



The default I2C addresses that the OLED and TMP102 modules have are fine.

Pushbuttons

At the minimum, the firmware needs two pushbutton switches connected between ground and GPIO 19 and GPIO 23 on the ESP board. They are the **+** and **-** buttons for the menu.

Another pushbutton switch can be connected between ground and GPIO 18 to display the date with a short press and the temperature with a long press (if a TMP102 module is installed). This switch is optional. In the menu the date and temperature can be set to display automatically for a few seconds each minute, too. See **Item 12 - Automatic Date and Temperature Display** on Page 40.

If Lixie II displays are used, a short double press on this switch toggles lighting their eleventh panels. The default is no lighting if the switch isn't installed.

Also a push button switch can be connected between ground and EN on the ESP board to reset the ESP. However most ESP modules come with built-in reset buttons. The switch is optional but it comes in handy if the ESP's reset switch isn't accessible.

To make using the menu easier, the switches should be tactile buttons so they have a definite click when pushed. Standard pushbuttons also work, but it's harder to do double presses.

Potentiometer

A 1000-ohm potentiometer can be connected to GPIO 34 (one of the ESP's ADC inputs).This is also optional. It sets the display color in manual color mode. However there is another way to do it if the potentiometer isn't installed. See **Item 17 - Automatic Color** on Page 45.

LED Data

DATA IN for the first edge-lit display connects to GPIO 13 on the ESP board. This is the leftmost display for NixieCron and EleksTube and the rightmost display for Lixie. Level conversion must be used as shown in the schematic diagrams.

Colons

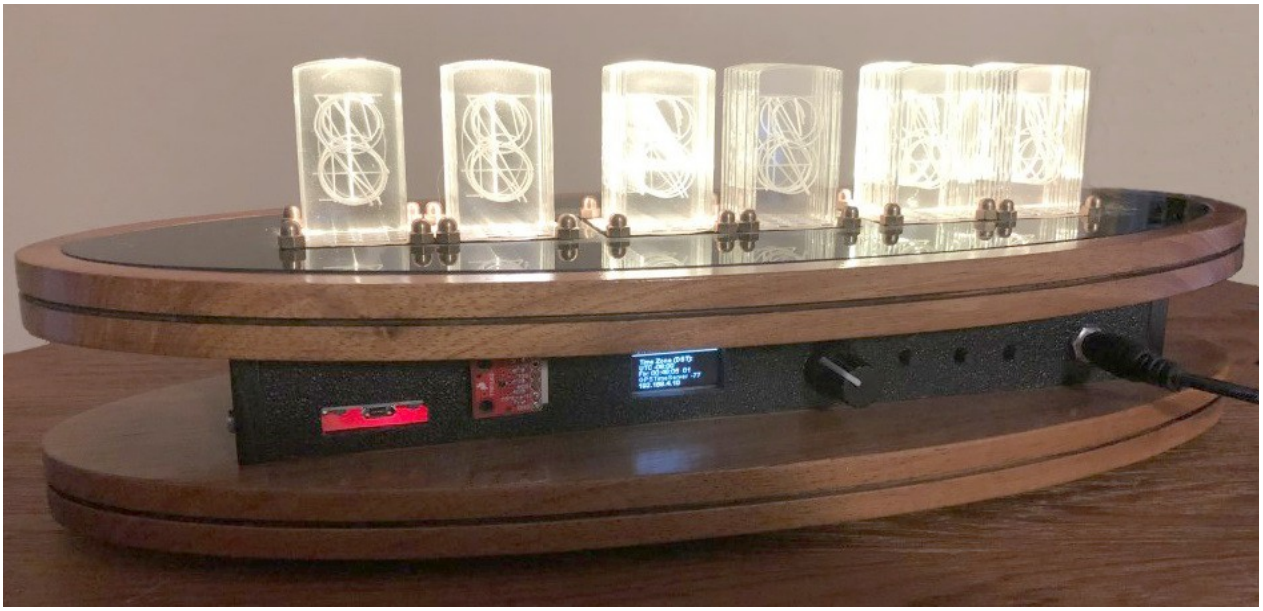
The firmware has provision for colons. These are two WS2812B LEDs connected to DATA OUT on the final edge-lit display in the clock. They are meant for edge-lit acrylic panels engraved with colons similar to the numeric displays, that's why there is only one LED per colon. The colon LEDs don't have to be installed for the firmware to work.

The menu selects whether the colons are off, on continuously, breathing (fading up and down each second), or flashing on and off each second. See **Item 4 - Colon Mode** on Page 32.

The colon LEDs should be connected as shown on the schematic with the first one located between the minutes and seconds and the other one located between the hours and minutes. Otherwise the modes that only need four digits (temperature and four-digit time/date) won't look right.

Here is an example of a clock running this firmware with Lixie displays and colons.

Here is an example with NixieCron “S” displays. This back view shows access to the ESP32 module's USB connector and reset button, the TMP102 module, the OLED display, the potentiometer, and three tactile pushbuttons.



UTC

The time that the clock receives from a Network Time Protocol (NTP) server and syncs with is Coordinated Universal Time. It's the world standard on which all time zones are based. Coordinated Universal Time is usually abbreviated UTC because of compromises made between English and translations to other languages. It's also known as Zulu Time in aviation, nautical, and military applications.

UTC is not a time zone. However for most practical purposes it's the same as the previous world standard, a time zone known as Greenwich Mean Time. It's a zone that runs through the United Kingdom, part of western Europe, and part of western Africa. Therefore in other time zones the hours, and in some cases also the minutes, that local time differs from UTC has to be taken into account.

The clock can automatically find the local UTC offset and it can also be set manually in the menu. These functions are described in **Automatically Setting the Local Time Zone** on Page 17 and the three menu items starting with **Item 8 - Manual Offset Hour** on Page 36.

Boot Sequence

Connecting to WiFi Network

First the version is shown in gray on the edge-lit display. It also shows on the OLED display. There is a two-second delay then the clock connects to the WiFi network. The edge-lit display shows an **orange** "**1**" and the OLED reads "**Connecting to WiFi...**".

There is a slight delay while connecting to the WiFi network. After connecting to the network the edge-lit display shows a **magenta** "**2**" and the OLED reads:

"Connected to" followed by the network's name

"**Setting time ...**"

If **six red zeros** are shown instead of the **magenta "2"** and the clock doesn't continue to the next step it means the clock couldn't connect to the network. See **Previous Network is Not Available or No Network Has Been Set** on Page 19.

Connecting to Time Server and Setting UTC Time

Once the clock connects to the WiFi network there is another slight delay while getting the time from the server. Usually the time sync is successful and the edge-lit display then shows a **cyan** "**3**" and the OLED reads:

"**UTC time set.**"

"**Setting time zone ...**"

If the clock couldn't sync to the time server it will still start but the time won't be right. In that case the edge-lit display shows a **red 3** and the OLED reads:

"**UTC time not set.**"

"**Setting time zone ...**"

If this happens check that the time server has been set properly. The easiest way is by viewing the information shown on the OLED display fo**r Item 33 - Network Information** (see Page 64). If the server is OK then just restart the clock and try again. If the server has to be set or changed follow the procedure described in the **WiFi** section on Page 19.

Automatically Setting the Local Time Zone

After UTC setup the clock moves on to setting the geographic information. The clock tries to find out where it's located by calling **"ipapi.co"**, a server on the Internet that returns the the location based on a WiFi network's IP address. It also returns the time zone which is needed to convert the UTC network time that the clock receives to local time.

The clock tries up to three times to contact the geographic server. A **blue** "**4**" is shown on the edge-lit display along with a number from 1 to 3 indicating which connection attempt Is currently happening. Usually this happens quickly on the first try but it could take longer.

If the display briefly shows a **blue** "**0**" instead of a number from 1 to 3 it means the automatic function has been disabled so there were no tries to contact the server (see menu **Item 7 - Automatic Time Zone**Page 35).

After connecting to the server and getting the information the edge-lit display shows a **green** "**5**" for 2½ seconds followed by the clock display. The OLED reads "**Time Zone set.**" for 2½ seconds then goes blank until it's needed for the menu.

Unable to Set the Time Zone Automatically

After three unsuccessful attempts to get the geographic information the clock uses the UTC offset that has been manually set in the menu. See the three menu items starting with **Item 8 - Manual Offset Hour** on Page 36.

In this case after the blue 1-2-3 sequence the edge-lit display shows a **yellow** "**5**" for 2½ seconds followed by the clock display. The OLED reads: "**Time Zone set.**" for 2½ seconds then goes blank until it's needed for the menu. The **yellow "5"** also shows if the automatic function was disabled.

The geographic information could be unavailable for several reasons. The server might just be temporarily unavailable or there could be temporary problems connecting to it.

The WiFi network that is providing the NTP sync may not have Internet access. This could be a temporary problem, too. However, a private NTP server, such as one that gets its time reference from GPS and doesn't have Internet access, can't reach the online geographic server. In that case the UTC offset will have to be set manually.

The geographic information or the manually set UTC offset can be shown on the OLED display. The easiest way to see it is by pressing the **Menu +** button while the clock is showing the time. See **Item 0 - Time Zone Information** on Page 28.

Boot Sequence Summary

The firmware version is displayed.

Orange 1 – The clock is attempting to connect to WiFi.

Six red zeros – The WiFi network previously set is not available or no network has been set. The clock is in server mode. Use a phone or computer to join the clock's “EdgeLit” network to set new WiFi credentials.

OR

Magenta 2 – WiFi connected successfully, now calling the time server.

Cyan 3 – UTC time was set.

OR

Red 3 – UTC time was not set.

Blue 4 along with a number from 1 to 3 – The clock is attempting to contact a geographic server to set the time zone. It tries up to three times indicated by the number following the **4**.

If a 0 shows instead of a number from 1 to 3 it means the automatic time zone function has been disabled in the menu.

Green 5 – The time zone was set automatically.

OR

Yellow 5 – The time zone could not be set automatically or the automatic function was disabled. The clock is using the UTC offset that was manually set in the menu instead.

Time and date display begin.

WiFi

There are two WiFi networks associated with this firmware. The first one is a home or commercial network that has Internet access. The ESP joins this network as a client. It connects to an NTP time server on the Internet and synchronizes the time every 5 minutes.

This network can also be provided by a private WiFi-enabled NTP server that doesn't have Internet access, such as a server that gets its time reference from GPS. However the automatic time zone feature won't work in this case. See **Unable to Set the Time Zone Automatically** on Page 17.

The ESP can also run in server mode with its own WiFi network to serve a web site where the credentials for the client network can be entered. The ESP only runs in server mode long enough for the information to be entered. After that it just connects to the client network and sticks to receiving time syncs and generating the data for the LEDs.

Previous Network is Available

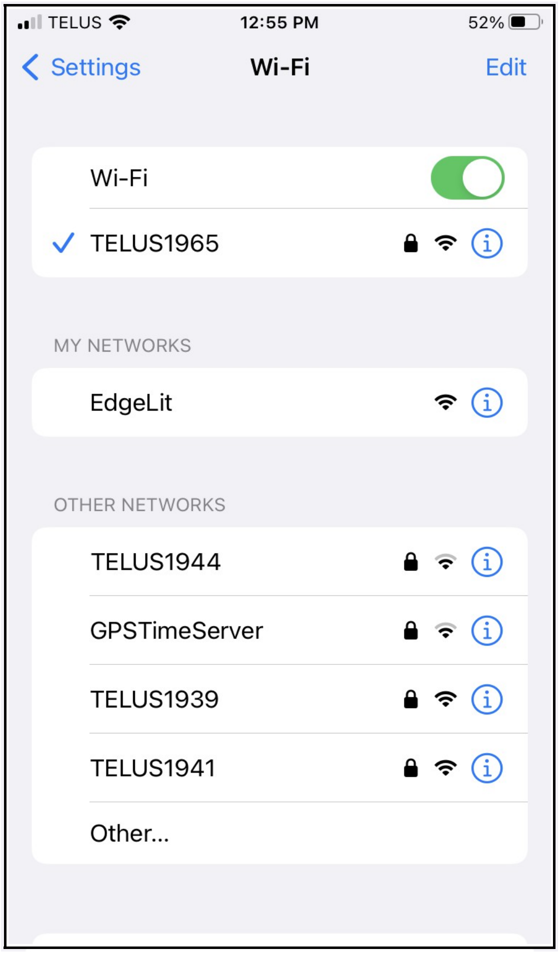
When the clock boots up it tries to connect as a client on the network that was used previously. After connecting the boot sequence continues normally.

Previous Network is Not Available or No Network Has Been Set

If the previous network isn't available or there was no previous network set, after showing the **orange** "**1**" on the edge-lit display, **six red zeros** are shown. This is when the ESP goes into server mode and sets up its own WiFi network called "**EdgeLit**" so you can enter the credentials for the client network that you want the clock to use.

The clock's **EdgeLit** network can be joined from a nearby WiFi enabled device such as a phone or a computer. No password is required.

Here is a screenshot showing the WiFi setup screen on an iPhone. It shows that the phone is currently connected to a network called TELUS1965 but the clock's **EdgeLit** network is there waiting to be joined.



TELUS 12:55 PM 52%

< Settings Wi-Fi Edit

Wi-Fi

✓ TELUS1965

MY NETWORKS

EdgeLit

OTHER NETWORKS

TELUS1944

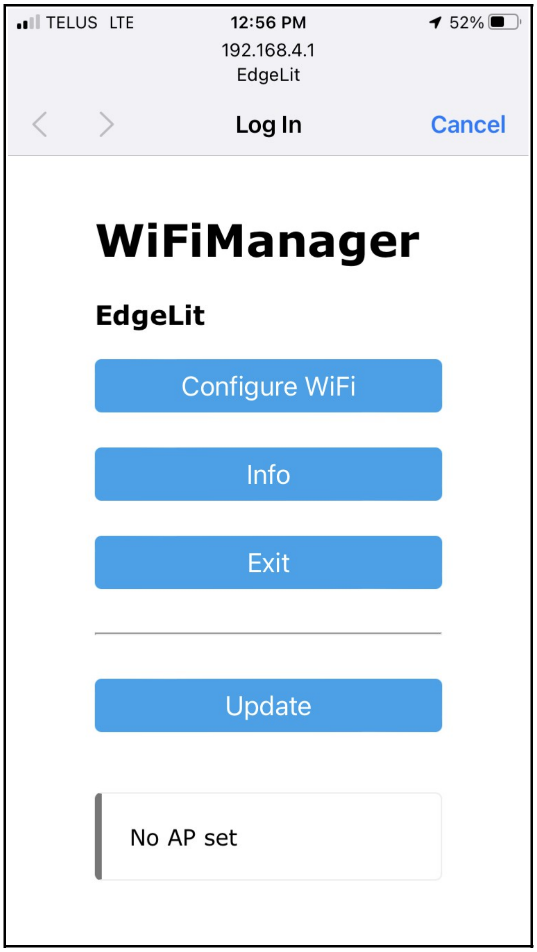
GPSTimeServer

TELUS1939

TELUS1941

Other...

The web site served is what's known as a "captive portal" so once the **EdgeLit** network is joined the phone or computer's web browser should automatically open and show this starting page.



WiFiManager

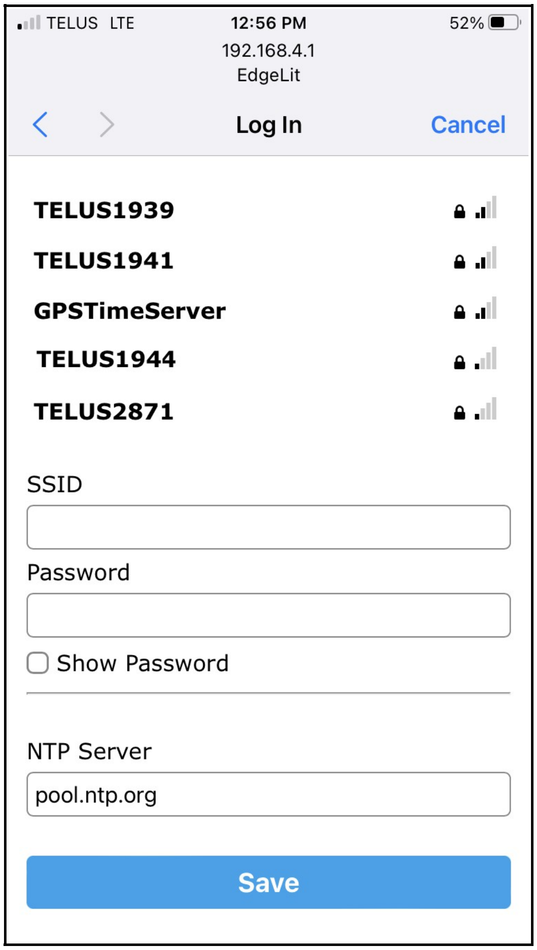
EdgeLit

No AP set

Captive portals usually work with phones but they may not work with computers. In case the web site doesn't open automatically just open the phone or computer's web browser and enter **192.168.4.1** in the address bar. This is the default IP address that ESP devices use in server mode.

There are several buttons on the page. Click on "**Configure WiFi**".

Another page opens showing all the available WiFi networks. Click on the one that you want the clock to use and enter the password in the text box labeled "**Password**". There is a checkbox where you can select whether the password is visible or not.



TELUS LTE 12:56 PM 52%

192.168.4.1
EdgeLit

Log In Cancel

TELUS1939

TELUS1941

GPSTimeServer

TELUS1944

TELUS2871

SSID

Password

□ Show Password

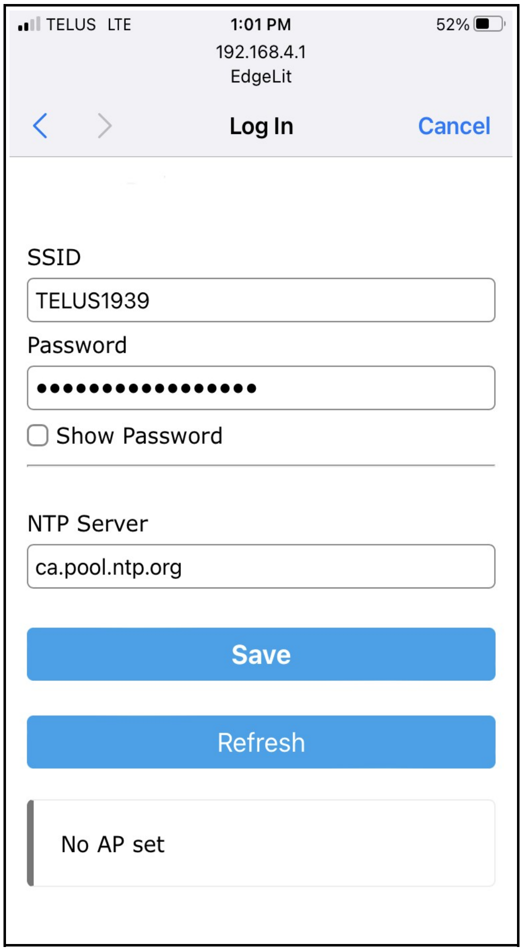
NTP Server

pool.ntp.org

Save

Setting the Time Server

There is another text box on the page labeled "**NTP Server**" where you enter the name or IP address for the time server that you want the clock to call.



SSID
TELUS1939

Password
•••••••••••••••••

☐ Show Password

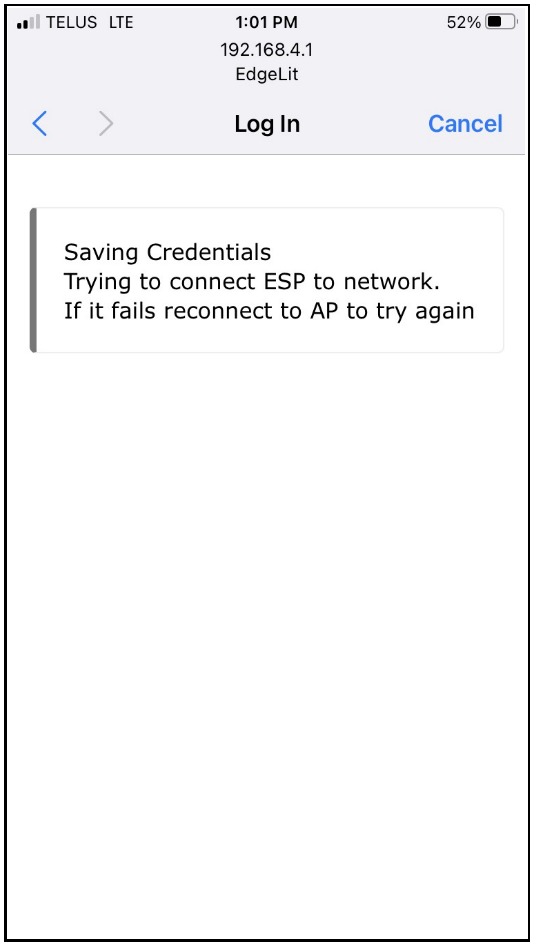
NTP Server
ca.pool.ntp.org

No AP set

By default the server is "**pool.ntp.org**". This should find a server from a pool of local Internet time servers. However there are numerous other time servers that can be called. For example if you want the pool for a specific country something like "**au.pool.ntp.org**" for Australia, "**ca.pool.ntp.org**" for Canada, or "**us.pool.ntp.org**" for the USA can be used. Or you could use "**time.google.com**", "**tick.usno.navy.mil**", "**time.nist.gov**", etc.

If you have your own WiFi-enabled NTP server, such as one that gets its time reference from GPS, you can select its network name from the list and enter its password. If its time server has a URL you can enter that in the NTP Server field. If not just enter the time server's IP address.

Click on "**Save**" after the WiFi network and time server have been entered. The following screen appears, then the ESP shuts down its **EdgeLit** network.



Saving Credentials
Trying to connect ESP to network.
If it fails reconnect to AP to try again

There is a slight delay while the ESP joins the chosen network as a client, then the second, third, and fourth parts of the boot sequence continue normally.

Changing the Network and Time Server

As mentioned earlier, when the clock boots up it will try to join the network that it was connected to previously. If that network isn't available the clock goes into server mode so you can enter the credentials for a new network. But if the previous network *is* available and you want to change it, or you want to change the time server, hold the **Menu -** button closed and reboot the clock.

The usual **orange** "**1**" shows on the edge-lit display indicating that the ESP is attempting to join a WiFi network. However now the ESP's existing WiFi network credentials have been deleted so it can't join any network. Therefore after showing the "**1**" for a couple of seconds the display shows the **red zeros** indicating that the clock's **EdgeLit** network is available and it's waiting for you to set a new WiFi network that it can join as a client to get the time.

You can release the **Menu -** button once the **red zeros** appear and just follow the same procedure described in **Previous Network is Not Available or No Network Has Been Set** on Page 19 to set the new network and time server.

Fails to Connect to the Network

If the clock doesn't connect to an existing network or time server and it should have, just reboot the clock and try again. Sometimes WiFi networks and servers are temporarily unavailable. Or sometimes digital devices just get confused and rebooting them cures a multitude of problems. Either press the ESP's **Reset** button or cycle the power to the clock.

After going into server mode and showing the **red zeros** for three minutes, the ESP will restart and attempt to connect to the previous WiFi network once again. This is handy when recovering from a power failure and it takes the WiFi router a minute or two to boot back up. In that case, the clock won't find the WiFi network because it boots up before the network is ready. Just let it be and the clock should start back up on its own.

Time Sync Indication

The firmware is set to sync with the time server every 5 minutes. Between checks the ESP uses its own timing functions to keep time.

The time when the most recent time fix arrived from the server is shown on the OLED and the edge-lit displays. The easiest way to see it is by pressing the **Menu -** button while the clock is showing the time. The OLED also shows information about the WiFi network and the time server. See **Item 33 - Network Information** on Page 64.

The blue LED on the ESP module indicates that the clock has synced with the time server. It goes off 10 seconds before the next sync is expected and gets re-lit when the sync happens. If the sync doesn't happen the LED will stay off. So as long as the blue LED is lit and never stays off for more than 10 seconds every 5 minutes, everything is working properly.

The blue LED on most ESP32 modules is wired so it is “active high”. That means the LED is connected between the GPIO 2 pin and ground and the ESP has to write a “1” to the pin to turn the LED on. However with some modules this is reversed and the LED connects between the GPIO 2 pin and +3.3 volts. This makes it “active low” and the the ESP has to write a “0” to the pin to turn the LED on.

This is accounted for in the firmware with a compiler directive. The directive is defined for active high since that's how the blue LED is wired on most ESP32 modules.

If the clock is syncing with the time server and the blue LED is off most of the time and only comes on for ten seconds every five minutes, it's because the LED is wired active low. To get it working properly the compiler directive has to be changed.

If using the Arduino IDE, open the **definitions.h** file and change the line that reads **#define LED_ON HIGH** to **#define LED_ON LOW**.

PlatformIO can use different “environments” so the same code can work with many different microcontroller modules. To change this compiler directive, open the **platformio.ini** file and locate the environment for the board you want to change. Then change the line that reads

build_flags = -D LED_ON=HIGH to **build_flags = -D LED_ON=LOW**

As usual the syntax for this, and any other statement in the code, is critical

Menu

The menu buttons respond to short presses, long presses (at least 800 milliseconds (about three quarters of a second), and double short presses.

The menu is invoked with a short press on either of the menu buttons. There are 34 items. The items and their settings appear on the edge-lit display and also in more detail on the OLED display. Since the OLED display can show text it makes using the menu very easy compared to the edge-lit displays which can only show numbers.

A short press on the **+** button steps through the items forward. A short press on the **-** button steps through the items backward. A double press on either button sets the edge-lit display back to normal and blanks the OLED. This also happens automatically if there are no button pushes for 15 seconds.

Once the menu item to be changed is showing, a long press on the **+** or **-** buttons changes the setting. Continuing to hold the buttons down won't do anything. A separate long press is required to step through each setting. After each long press the the OLED and edge-lit displays are updated to show the new setting. The menu item will continue to be selected and ready to accept further changes until any of the following happen:

-Another item is selected with a short press on either menu button

-The menu is closed with a double short press on either menu button

-The clock returns to time display on its own after 15 seconds with no button presses

When the menu exits, either with a double press on one of the menu buttons or when it times out on its own, any settings that were changed will be saved in EEPROM so they don't have to be set each time the clock is restarted.

Here are the menu items and what the edge-lit display and OLED display say about them. The default settings are indicated by an asterisk (*).

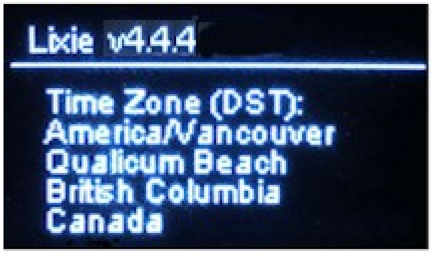
Item 0 - Time Zone Information

This is the first thing shown when the menu is invoked by pressing the **+** button. This is information only - nothing can be changed here. However the UTC offset can be changed using the three menu items starting with **Item 8 - Manual Offset Hour** on Page 36.

OLED:

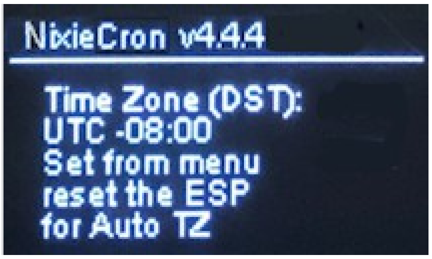
Display type and firmware version

Time zone name, daylight saving time status, city, region, and country are shown if they are available



Lixie v4.4.4
Time Zone (DST):
America/Vancouver
Qualicum Beach
British Columbia
Canada

If geographic information is not available or is not used daylight saving time status and the UTC offset that was set in the menu are shown.
Default is Pacific Standard Time (UTC -08:00).



NixieCron v4.4.4
Time Zone (DST):
UTC -08:00
Set from menu
reset the ESP
for Auto TZ

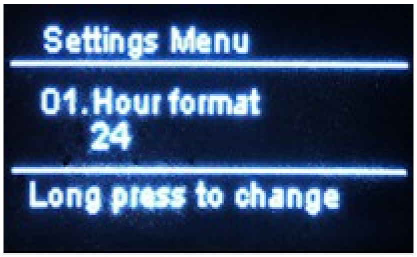
Edge-lit display:

The offset hours are shown on the third and fourth displays, the offset minutes are shown on the fifth and sixth displays. A “0” shows on the first display if the offset is negative (behind UTC).

Item 1 - Hour Format

Choices:
24 hour time *
12 hour time

OLED:
"Hour format:" followed by "24" or "12"
This is typical of how all menu settings are displayed



Settings Menu

01.Hour format
24

Long press to change

Edge-lit display:
"01" followed by "0" if 12 hour or "1" if 24 hour

A long press on either menu button toggles the two states

Item 2 - Leading Zero Blanking for Time and Date

A zero can be shown on the first edge-lit display in front of single-digit hours, days, and months if desired.

Choices:

Enable it so the leading zero is blanked *

Disable it so the leading zero is displayed

OLED:

"Lead zero blanking:" followed by "Enabled" or "Disabled"

Edge-lit display:

"02" followed by "0" if disabled or "1" if enabled

A long press on either menu button toggles the two states

Item 3 - Digits

Disables seconds and year display if desired.

Choices:

Four - the last two digits on the six-digit display aren't used but they will show the aura on inactive digits in Nixie mode

Six - normal display *

OLED:

"Digits:" followed by "Four" or "Six"

Edge-lit display:

"03" followed by "0" if four or "1" if six

A long press on either menu button toggles the two states

Item 4 - Colon Mode

Choices:

Off

Steady - on all the time

Breathing - fades up and down each second *****

Flashing - flashes on and off each second

OLED:

"Colons:" followed by "Off", "Steady", "Breathing" or "Flashing"

Edge-lit display:

"04" followed by "0" if off, "1" if steady, "2" if breathing, or "3" if flashing

A long press on **+** goes through the choices forward

A long press on **-** goes through the choices backward

Item 5 - Colon Minimum Brightness

How bright the colons are at the bottom of the breathing effect.

The firmware doesn't allow values equal to or greater than the maximum value that's currently set.

Choices:

0 to 255 (**default 30**)

OLED:

"Colon min brightness:" followed by the numeric value

Edge-lit display:

"05" followed by the numeric value

A long press on **+** increments the value in 15-unit steps

A long press on **-** decrements the value in 15-unit steps

Item 6 - Colon Maximum Brightness

How bright the colons are in all modes.

Setting maximum brightness equal to the currently set minimum value will automatically reduce the minimum as well.

Choices:

0 to 255 (**default 120**)

OLED:

"Colon max brightness:" followed by the numeric value

Edge-lit display:

"06" followed by the numeric value

A long press on **+** increments the value in 15-unit steps

A long press on **-** decrements the value in 15-unit steps

Item 7 - Automatic Time Zone

Sets whether the clock tries to find the time zone automatically from a geographic server on the Internet or if it just uses the UTC offset that was manually set from the menu. Disable the automatic function if you want the clock to always show the time from somewhere other than where it's located.

Also disabling the automatic function speeds up the boot sequence if the geographic server isn't available. That way the clock will always use the manual settings rather than trying unsuccessfully to find the server before finally reverting to the manual settings (see **Unable to Set the Time Zone Automatically** on Page 17).

Choices:

Enable it to use the automatic function *

Disable it to use the manual settings

OLED:

"Auto TZ:" followed by "Enabled" or "Disabled"

Edge-lit display:

"07" followed by "0" if disabled, or "1" if enabled

A long press on either menu button toggles the two states

If this item is changed the clock has to be restarted for it to take effect.

The UTC offset is normally received automatically from **ipapi.co**. If it's not available the default UTC offset is used (Pacific Standard Time, unless this has been changed with these three menu items).

The clock uses the manual offset any time these items are changed. Reset the ESP to go back to automatic time zone offset.

The clock can be set to always use manual offset if desired. See **Item 7 - Automatic Time Zone** on Page 35.

Set the UTC offset for Standard Time in the desired time zone.

The clock knows when DST should be in effect even if it can't receive data from **ipapi.co.** This is based on rules specified with the eight menu items starting with **Item 24 - DST Start Week.** Automatic DST adjustments can be disabled with **Item 23 - Automatic DST** if your location doesn't use it. See Pages 51 to 59 for these settings.

See Page 15 for more information about UTC.

Item 8 - Manual Offset Hour

This item works in conjunction with **Item 9 - Manual Offset Minute** to set the UTC offset manually. Don't worry about negative numbers (offsets behind UTC) here. That's accounted for in **Item 10 - Manual Offset Polarity**.

Choices:

0 to 14 (**default 8**)

OLED:

"Manual offset hour:" followed by the numeric value

Edge-lit display:

"08" followed by the numeric value

A long press on **+** increments the value in 1-hour steps

A long press on **-** decrements the value in 1-hour steps

Item 9 - Manual Offset Minute

This item works in conjunction with **Item 8 - Manual Offset Hour** to set the UTC offset manually.

Some places in the world have time zones 30 or 45 minutes different from adjacent areas, such as India, Nepal, Iran, Afghanistan, parts of Australia, and the Canadian province of Newfoundland.

Choices:

0 to 45 (**default 0**)

OLED:

"Manual offset minute:" followed by the numeric value

Edge-lit display:

"09" followed by the numeric value

A long press on **+** increments the value in 15-minute steps

A long press on **-** decrements the value in 15-minute steps

Item 10 - Manual Offset Polarity

This item works in conjunction with **Item 8 - Manual Offset Hour** and **Item 9 - Manual Offset Minute** to set the UTC offset manually.

It specifies whether the offset is ahead or behind UTC.

Choices:

Behind UTC (-) *

Ahead UTC (+)

OLED:

"Man. offset polarity:" followed by "Behind UTC (-)" or "Ahead UTC (+)"

Edge-lit display:

"10" followed by "0" if polarity is negative or "1" if polarity is positive

A long press on either menu button toggles the two states

Item 11 - Date Format

Choices:
Month Day Year *
Day Month Year
Year Month Day

OLED:
"Date format:" followed by "MM DD YY", "DD MM YY" or "YY MM DD"

Edge-lit display:
"11" followed by "0" if Month Day Year, "1" if Day Month Year, or "2" if Year Month Day

A long press on Menu + goes through the choices forward
A long press on Menu - goes through the choices backward

Item 12 - Automatic Date and Temperature Display

Choices:

Enable it so temperature and date display automatically for four seconds each minute. Date is shown between 36 and 39 seconds. Temperature is shown between 6 and 9 seconds (if the TMP102 module is installed). *

Disable it so only the time is displayed

OLED:

"Date/Temperature:" followed by "Time only" or "Auto"

Edge-lit display:

"12" followed by "0" if time only, or "1" if auto

A long press on either menu button toggles the two states

Item 13 - Temperature Units

Choices:

Fahrenheit

Celsius *

OLED:

"Temperature units:" followed by "Fahrenheit" or "Celsius"

Edge-lit display:

"13" followed by "0" if Fahrenheit, or "1" if Celsius

A long press on either menu button toggles the two states

Item 14 - Nixie Mode

This firmware can simulate some of the way Nixie tubes look by using specific colors for the active and inactive edge-lit digits. The active digits show an amber “ionization” color and the inactive digits show a pale purplish-blue “aura” color. The brightness of the aura can be set with **Item 15 - Nixie Aura Intensity** (see Page 43). Nixie clocks also periodically show random numbers to keep all the digits exercised. This is simulated with **Item 16 - Random Mode** (see Page 44)**.**

Choices:

Enable it to show the Nixie simulation - overrides all other color settings *****

Disable it to show color modes selected in **Item 17 - Automatic Color** and **Item 18 - Color Variation**

OLED:

"Nixie mode:" followed by "Enabled" or "Disabled"

Edge-lit display:

"14" followed by "0" if disabled, or "1" if enabled

A long press on either menu button toggles the two states

Item 15 - Nixie Aura Intensity

How bright the background color is in Nixie mode (the purplish-blue “aura” on the unlit digit panels). It's a subtle effect and shouldn't be set too high to look authentic.

Choices:

0 to 15 (**default 2**)

OLED:

"Nixie aura intensity:" followed by the numeric value

Edge-lit display:

"15" followed by the numeric value

A long press on **+** increments the value in 1-unit steps

A long press on **-** decrements the value in 1-unit steps

Item 16 - Random Mode

Simulates anti-cathode poisoning routines for Nixie tubes by showing random numbers on the edge-lit displays for 10 seconds every 10 minutes.

Happens during 2, 12, 22, 32, 42, and 52 minutes. Starts 15 seconds past the minute so it doesn't interfere with automatic date and temperature display.

Choices:

Enable it to allow the random numbers *

Disable it to not allow random numbers

OLED:

"Random:" followed by "Enabled" or "Disabled"

Edge-lit display:

"16" followed by "0" if disabled, or "1" if enabled

A long press on either menu button toggles the two states

Background Information for Random Mode

Before LED readouts were common, Nixie tubes were the main numeric displays for equipment in the 1950s and 1960s. They have an anode screen that surrounds ten cathodes all inside a sealed glass tube containing neon gas. Each cathode is shaped like one of the digits from 0 to 9. A digit is turned on by applying a relatively high positive DC voltage to the anode and grounding one of the cathodes. This ionizes the neon gas surrounding the cathode causing it to emit light.

If one cathode is lit for long periods of time while the others are inactive, as in a Nixie tube showing the hours on a clock, a resistive coating builds up on the inactive cathodes. This is known as "cathode poisoning". If the coating on a cathode gets thick enough, parts of it will glow dimmer or be totally dark if it is switched on. That's why Nixie clocks have routines to exercise all the cathodes at regular intervals to help prevent them from getting "poisoned".

This firmware can simulate some of the way Nixie tubes look by using specific colors for the active and inactive edge-lit digits. LEDs don't need anti-cathode poisoning routines, of course, the code to do it is just here for authenticity. Also it's kind of a neat looking effect.

Item 17 - Automatic Color

Choices:

Static, manual - steady color manually selected with the potentiometer

Changing, auto - gradually cycles through all the hues *

OLED:

"Auto color:" followed by "Static, manual" or "Changing, auto"

Edge-lit display:

"17" followed by "0" if static, or "1" if automatic

A long press on either menu button toggles the two states

The potentiometer is optional. If it's not there the static color is "Nixie amber" by default. However it will stay whatever color the display is currently showing when the mode is changed from auto to manual. So leave it set to auto until the color you want is showing then change it to manual.

Item 18 - Color Variation

Choices:

All digits the same color *

Rainbow where each pair of digits is a different color

OLED:

"Color variation:" followed by "All the same" or "Rainbow pairs"

Edge-lit display:

"18" followed by "0" if all the same or "1" if rainbow pairs

A long press on either menu button toggles the two states

Item 19 - Day Brightness

How bright the edge-lit display is during day hours.

Choices:
0 to 255 (**default 255**)

OLED:
"Day brightness:" followed by the numeric value

Edge-lit display:
"19" followed by the numeric value

A long press on **+** increments the value in 15-unit steps
A long press on **-** decrements the value in 15-unit steps

Item 20 - Night Brightness

How bright the edge-lit display is during night hours specified in **Item 21 - Night Begin Hour** and **Item 22 - Night End Hour.**

Choices:

0 to 255 (**default 30**)

OLED:

"Night brightness:" followed by the numeric value

Edge-lit display:

"20" followed by the numeric value

A long press on **+** increments the value in 15-unit steps

A long press on **-** decrements the value in 15-unit steps

Item 21 - Night Begin Hour

When day mode stops and night mode begins.

Choices:
0 to 23 (**default 23**)

OLED:
"Night begin hour:" followed by the numeric value

Edge-lit display:
"21" followed by the numeric value

A long press on **+** increments the value in 1 hour steps
A long press on **-** decrements the value in 1 hour steps

Item 22 - Night End Hour

When night mode stops and day mode begins.

Choices:

0 to 23 (**default 8**)

OLED:
"Night end hour:" followed by the numeric value

Edge-lit display:
"22" followed by the numeric value

A long press on **+** increments the value in 1-hour steps
A long press on **-** decrements the value in 1-hour steps

The following items specify the rules for when Daylight Saving Time starts and ends in your location so the clock can adjust the time automatically. The default settings are for most of North America. Yes, we are still silly enough to bother with this nonsense!

If DST is not observed in your location disable the automatic function in **Item 23 - Automatic DST** and the values in the eight menu items beginning with **Item 24 - DST Start Week** don't matter.

Item 23 - Automatic DST

Automatically adjusts the time when Daylight Saving Time is in effect.

Choices:

Enable it to allow adjustments based on the following menu settings *

Disable it to not allow adjustments if DST is not observed in your location

OLED:

"Auto DST:" followed by "Enabled" or "Disabled"

Edge-lit display:

"23" followed by "0" if disabled, or "1" if enabled

A long press on either menu button toggles the two states

Item 24 - DST Start Week

The week of the month when Daylight Saving Time begins.

Choices:
Last
First
Second *
Third
Fourth

OLED:
"DST start week:" followed by "Last", "First", "Second", “Third”, or "Fourth"

Edge-lit display:
"24" followed by "0" if Last, "1" if First, "2" if Second, “3” if Third or "4" if Fourth

A long press on **+** goes through the choices forward
A long press on **-** goes through the choices backward

Item 25 - DST Start Day

The day of the week when Daylight Saving Time begins.

Choices:
Sunday *
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday

OLED:
"DST start day:" followed by “Sunday”, “Monday”, “Tuesday”, Wednesday”, “Thursday”, “Friday”, or “Saturday”

Edge-lit display:
"25" followed by "1" if Sunday, "2" if Monday, “3" if Tuesday, “4” if Wednesday, “5” if Thursday, “6” if Friday, or "7" if Saturday

A long press on **+** goes through the choices forward
A long press on **-** goes through the choices backward

Item 26 - DST Start Month

The month when Daylight Saving Time begins.

Choices:
January
February
March *
April
May
June
July
August
September
October
November
December

OLED:
"DST start month:" followed by the name of the month

Edge-lit display:
"26" followed by the number of the month (January is “1”, February is “2”, March is “3”, April is “4”, May is “5”, June is “6”, July is “7”, August is “8”, September is “9”, October is “10”, November is “11”, December is “12”)

A long press on **+** goes through the choices forward
A long press on **-** goes through the choices backward

Item 27 - DST Start Hour

The hour when Daylight Saving Time begins.

Choices:
0 to 23 (**default 2**)

OLED:
"DST start hour:" followed by the numeric value

Edge-lit display:
"27" followed by the numeric value

A long press on **+** increments the value in 1 hour steps
A long press on **-** decrements the value in 1 hour steps

Item 28 - DST End Week

The week of the month when Daylight Saving Time ends.

Choices:

Last

First *****

Second

Third

Fourth

OLED:

"DST end week:" followed by "Last", "First", "Second", “Third”, or "Fourth"

Edge-lit display:

"28" followed by "0" if Last, "1" if First, "2" if Second, “3” if Third or "4" if Fourth

A long press on **+** goes through the choices forward

A long press on **-** goes through the choices backward

Item 29 - DST End Day

The day of the week when Daylight Saving Time ends.

Choices:
Sunday *
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday

OLED:
"DST end day:" followed by “Sunday”, “Monday”, “Tuesday”, Wednesday”,
“Thursday”, “Friday”, or “Saturday”

Edge-lit display:
"29" followed by "1" if Sunday, "2" if Monday, "3" if Tuesday, “4” if Wednesday,
“5” if Thursday, “6” if Friday, or "7" if Saturday

A long press on **+** goes through the choices forward
A long press on **-** goes through the choices backward

Item 30 - DST End Month

The month when Daylight Saving Time ends.

Choices:

January

February

March

April

May

June

July

August

September

October

November *

December

OLED:

"DST end month:" followed by the name of the month

Edge-lit display:

"30" followed by the number of the month (January is “1”, February is “2”, March is “3”, April is “4”, May is “5”, June is “6”, July is “7”, August is “8”, September is “9”, October is “10”, November is “11”, December is “12”)

A long press on **+** goes through the choices forward

A long press on **-** goes through the choices backward

Item 31 - DST End Hour

The hour when Daylight Saving Time ends.

Choices:
0 to 23 (**default 2**)

OLED:
"DST end hour:" followed by the numeric value

Edge-lit display:
"31" followed by the numeric value

A long press on **+** increments the value in 1 hour steps
A long press on **-** decrements the value in 1 hour steps

Item 32 - Serial Monitor

If the ESP is connected to a computer all the menu settings, network information, and general status can be shown on a serial monitor. Connect via USB and run terminal software (such as PuTTY or the serial monitor in the PlatformIO or Arduino IDEs). Set the terminal's port to whichever one the ESP is connected to and the speed to **115200**.

This is mainly debugging information that was useful when developing the code. However it's still here to show that everything is working properly.

Choices:

Detailed - Shows all the menu settings at once along with the time, date, temperature, and the geographic, network, and time sync information. The information updates every 5 seconds and when any switch is pressed.

Minimal - Only information about the sync event and the time server is shown when the clock syncs with the server every 5 minutes. This helps to conserve CPU resources when nobody is watching the serial monitor. However generating and updating the detailed information is not a problem with an ESP32. *

OLED:

"Serial monitor” followed by "Detailed" or "Minimal"

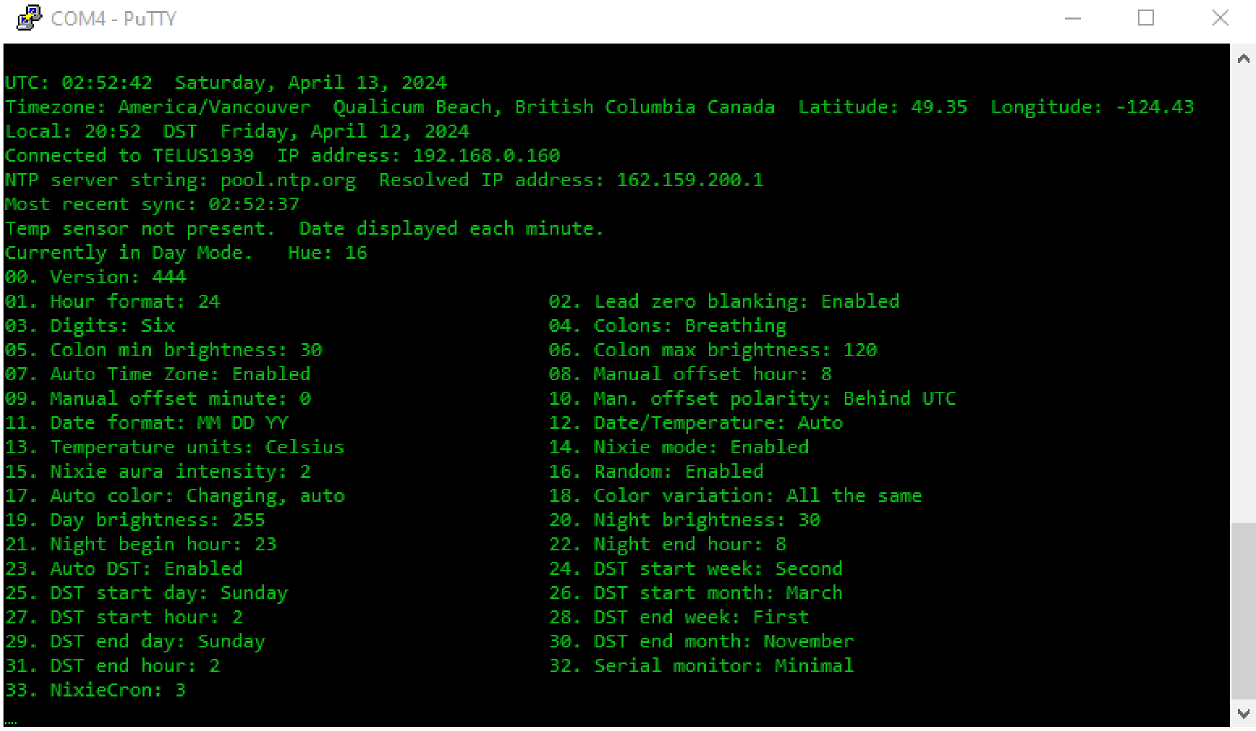
Edge-lit display:

"32" followed by "0" if minimal, or "1" if detailed

A long press on either menu button toggles the two states

This screenshot shows the detailed information. The clock is connected to a WiFi network called **TELUS1939** and its IP address on that network is **192.168.0.160**. The network has Internet access so the clock received the UTC offset along with other information from the geographic server.

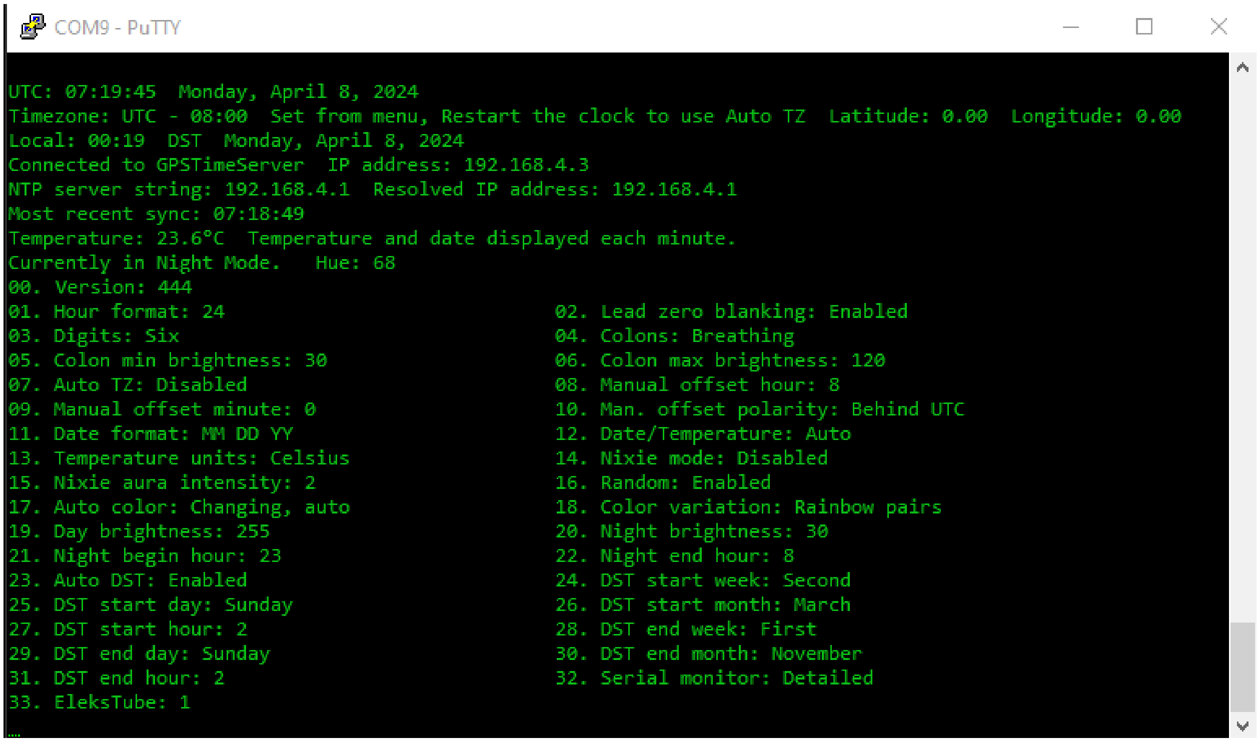
The **Timezone** line shows the zone and the location. The **NTP server string** line shows that the URL for the NTP server being called is **pool.ntp.org.** The firmware can resolve a URL's IP address and that's **162.159.200.1** here.



```
UTC: 02:52:42  Saturday, April 13, 2024
Timezone: America/Vancouver  Qualicum Beach, British Columbia Canada  Latitude: 49.35  Longitude: -124.43
Local: 20:52  DST  Friday, April 12, 2024
Connected to TELUS1939  IP address: 192.168.0.160
NTP server string: pool.ntp.org  Resolved IP address: 162.159.200.1
Most recent sync: 02:52:37
Temp sensor not present.  Date displayed each minute.
Currently in Day Mode.   Hue: 16
00. Version: 444
01. Hour format: 24                             02. Lead zero blanking: Enabled
03. Digits: Six                                 04. Colons: Breathing
05. Colon min brightness: 30                    06. Colon max brightness: 120
07. Auto Time Zone: Enabled                     08. Manual offset hour: 8
09. Manual offset minute: 0                     10. Man. offset polarity: Behind UTC
11. Date format: MM DD YY                       12. Date/Temperature: Auto
13. Temperature units: Celsius                  14. Nixie mode: Enabled
15. Nixie aura intensity: 2                     16. Random: Enabled
17. Auto color: Changing, auto                  18. Color variation: All the same
19. Day brightness: 255                         20. Night brightness: 30
21. Night begin hour: 23                        22. Night end hour: 8
23. Auto DST: Enabled                           24. DST start week: Second
25. DST start day: Sunday                       26. DST start month: March
27. DST start hour: 2                           28. DST end week: First
29. DST end day: Sunday                         30. DST end month: November
31. DST end hour: 2                             32. Serial monitor: Minimal
33. NixieCron: 3
```

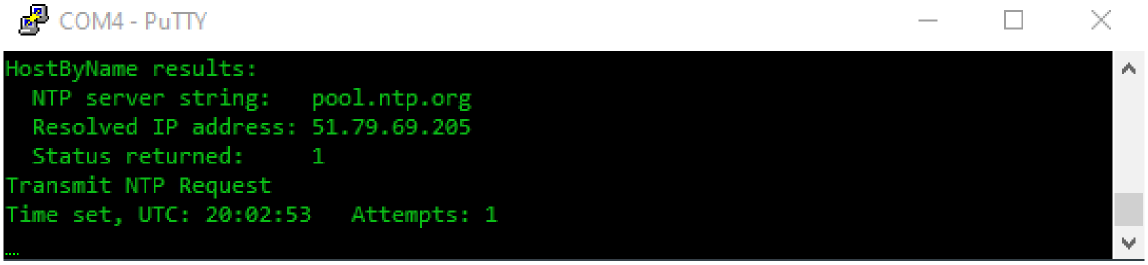
Here's another screenshot of the detailed information. This time the clock is connected to a private WiFi network called **GPSTimeServer** and its IP address on that network is **192.168.4.3**. This network gets the time from GPS so it doesn't have Internet access. Therefore the clock can't get the timezone information automatically and it's using the UTC offset that was manually set in the menu.

In this case, the **Timezone** line just shows the hours and minutes for the offset which is **-8:00**. This network's time server doesn't have a URL so the **NTP server string** line just shows the server's IP address which is **192.168.4.1.** The resolved IP address is the same.



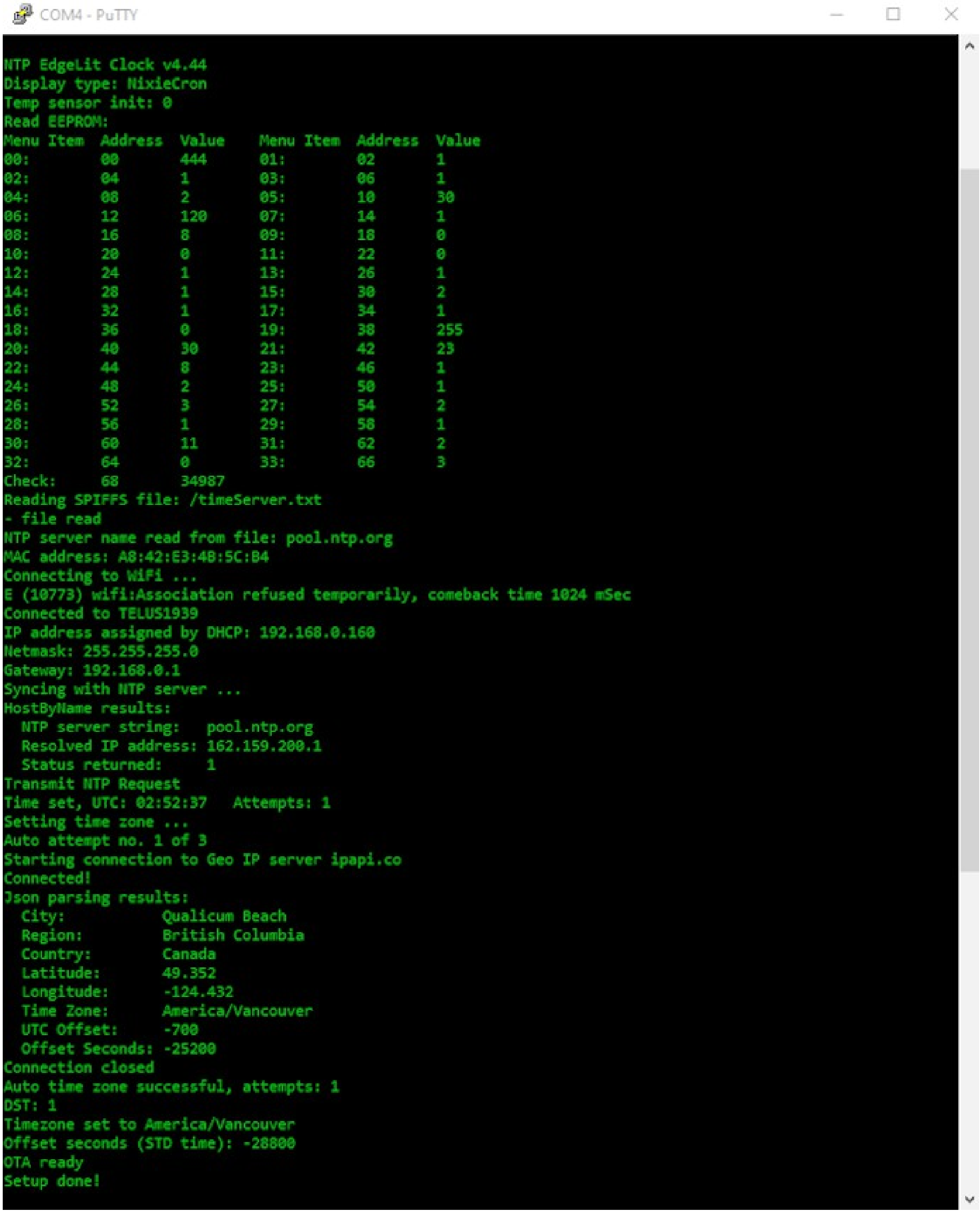
```
UTC: 07:19:45  Monday, April 8, 2024
Timezone: UTC - 08:00  Set from menu, Restart the clock to use Auto TZ  Latitude: 0.00  Longitude: 0.00
Local: 00:19  DST  Monday, April 8, 2024
Connected to GPSTimeServer  IP address: 192.168.4.3
NTP server string: 192.168.4.1  Resolved IP address: 192.168.4.1
Most recent sync: 07:18:49
Temperature: 23.6°C  Temperature and date displayed each minute.
Currently in Night Mode.   Hue: 68
00. Version: 444
01. Hour format: 24                             02. Lead zero blanking: Enabled
03. Digits: Six                                 04. Colons: Breathing
05. Colon min brightness: 30                    06. Colon max brightness: 120
07. Auto TZ: Disabled                           08. Manual offset hour: 8
09. Manual offset minute: 0                     10. Man. offset polarity: Behind UTC
11. Date format: MM DD YY                       12. Date/Temperature: Auto
13. Temperature units: Celsius                  14. Nixie mode: Disabled
15. Nixie aura intensity: 2                     16. Random: Enabled
17. Auto color: Changing, auto                  18. Color variation: Rainbow pairs
19. Day brightness: 255                         20. Night brightness: 30
21. Night begin hour: 23                        22. Night end hour: 8
23. Auto DST: Enabled                           24. DST start week: Second
25. DST start day: Sunday                       26. DST start month: March
27. DST start hour: 2                           28. DST end week: First
29. DST end day: Sunday                         30. DST end month: November
31. DST end hour: 2                             32. Serial monitor: Detailed
33. EleksTube: 1
```

This screenshot shows the information presented when a sync event happens every 5 minutes. If this menu item is set to **Minimal** this is all that shows on the serial monitor. It also shows if set to **Detailed**, along with all the information shown above.



```
HostByName results:
  NTP server string:   pool.ntp.org
  Resolved IP address: 51.79.69.205
  Status returned:     1
Transmit NTP Request
Time set, UTC: 20:02:53   Attempts: 1
```

If a computer is connected to the ESP during boot-up, lots of information about what's happening is shown regardless of this menu setting.



```
NTP EdgeLit Clock v4.44
Display type: NixieCron
Temp sensor init: 0
Read EEPROM:
Menu Item  Address  Value    Menu Item  Address  Value
00:        00       444      01:        02       1
02:        04       1        03:        06       1
04:        08       2        05:        10       30
06:        12       120      07:        14       1
08:        16       8        09:        18       0
10:        20       0        11:        22       0
12:        24       1        13:        26       1
14:        28       1        15:        30       2
16:        32       1        17:        34       1
18:        36       0        19:        38       255
20:        40       30       21:        42       23
22:        44       8        23:        46       1
24:        48       2        25:        50       1
26:        52       3        27:        54       2
28:        56       1        29:        58       1
30:        60       11       31:        62       2
32:        64       0        33:        66       3
Check:     68       34987
Reading SPIFFS file: /timeServer.txt
- file read
NTP server name read from file: pool.ntp.org
MAC address: A8:42:E3:4B:5C:B4
Connecting to WiFi ...
E (10773) wifi:Association refused temporarily, comeback time 1024 mSec
Connected to TELUS1939
IP address assigned by DHCP: 192.168.0.160
Netmask: 255.255.255.0
Gateway: 192.168.0.1
Syncing with NTP server ...
HostByName results:
  NTP server string:   pool.ntp.org
  Resolved IP address: 162.159.200.1
  Status returned:     1
Transmit NTP Request
Time set, UTC: 02:52:37   Attempts: 1
Setting time zone ...
Auto attempt no. 1 of 3
Starting connection to Geo IP server ipapi.co
Connected!
Json parsing results:
  City:           Qualicum Beach
  Region:         British Columbia
  Country:        Canada
  Latitude:       49.352
  Longitude:      -124.432
  Time Zone:      America/Vancouver
  UTC Offset:     -700
  Offset Seconds: -25200
Connection closed
Auto time zone successful, attempts: 1
DST: 1
Timezone set to America/Vancouver
Offset seconds (STD time): -28800
OTA ready
Setup done!
```

Item 33 - Network Information

This is the first thing shown when the menu is invoked by pressing the **Menu -** button. This is information only - nothing can be changed here. However the WiFi network and time server can be changed. See **Changing the Network and Time Server** on Page 25.

OLED:

Firmware version and display type

The WiFi network the clock is connected to (TELUS1939 in the photo)

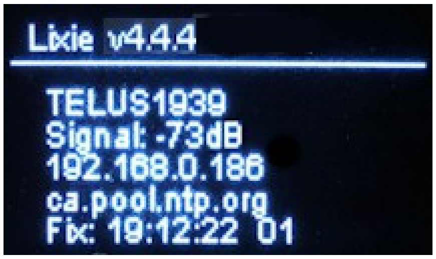
The signal strength (-73dB in the photo)

The clock's IP address (192.168.0.186 in the photo)

The time server being called (ca.pool.ntp.org in the photo)

The UTC time the most recent time fix arrived, they should come in every 300 seconds (5 minutes). It also shows how many tries it took to contact the time server.

(19:12:22 with one try in the photo)



```
Lixie v4.4.4
TELUS1939
Signal: -73dB
192.168.0.186
ca.pool.ntp.org
Fix: 19:12:22 01
```

Edge-lit display:

The time the most recent time fix arrived

Additional Button Functions

Restore Default Menu Settings

The default menu settings can be restored to the EEPROM by holding the **Menu +** button closed and resetting the ESP. The button can be released during the boot up sequence anytime after the version shows on the displays.

Change WiFi Network or Time Server

The WiFi network and the server the clock uses to get the time can be changed by holding the **Menu -** button closed and resetting the ESP. See **Changing the Network and Time Server** on Page 25 for details.

Date/Temperature Button

The Date/Temperature button invokes continuous display of these items regardless of the auto setting in the menu.

A short press shows the date. Another short press changes back to time.

A long press shows the temperature. Another long press changes back to time.

The other modes can be invoked no matter which mode is currently displayed. IE., to show the date it's a short press whether time or temperature is showing; if date is showing it's a short press to go back to time.

To show the temperature it's a long press whether time or date is showing; if temperature is showing it's a long press to go back to time.

A long press does nothing if the TMP102 board isn't installed (see **Error: Reference source not found** on Page Error: Reference source not found).

A double short press toggles illumination of the **eleventh panel on Lixie II** displays. It has no effect on other displays.

Appendix

Quick Start First Time Setup Guide

Connect the clock to a 5-volt power supply.

The version shows on the edge-lit displays in gray, followed by an **orange** "**1**",

then **six red zeros**.

The clock now has a WiFi network named "**EdgeLit**" running waiting for user input. Connect to this network using any nearby phone or computer. No password is needed.

A web site called "**WiFi Manager EdgeLit**" should open on the phone or computer's browser. If it doesn't, open the browser and enter **192.168.4.1** in the address bar.

When the page opens click on the "**Configure WiFi**" button. Another page opens showing all the available WiFi networks.

Click on the network you want the clock to use

Enter the network's password in the text box labeled "**Password**".

Enter the time server you want the clock to use in the text box labeled "**NTP Server**".

The default is "**pool.ntp.org**" which should find a server from a local pool of servers.

You can enter the name of a different server if you want, such as **time.google.com**, or just enter the server's IP address.

Click on "**Save**".

The clock shuts down its "**EdgeLit**" network and connects to the chosen WiFi network.

A **magenta** "**2**" shows on the edge-lit displays after successfully connecting to the WiFi network and while the clock calls the time server and sets the UTC time.

A **cyan** "**3**" shows on the edge-lit displays after successfully setting the UTC time.

A **blue** "**4**" shows on the edge-lit displays while setting the time zone. The clock tries three times to do this automatically by contacting a geographic server. The current attempt is indicated by the number following the “**4**”. A **green** "**5**" shows on the edge-lit displays if the clock was able to set the time zone automatically.

A **yellow** "**5**" shows if it couldn't set the time zone and it's using the default UTC offset.

Normal time display begins.

If the time isn't right check the UTC offset settings in the menu. See the three menu items starting with **Item 8 - Manual Offset Hour** on Page 36.

If any of this didn't happen, reboot and try again. Sometimes WiFi can be temperamental.

After successfully setting this up, the next time the clock is started the boot sequence will skip the **red zeros** part and simply connect to the previously chosen WiFi network and time server and start showing the time.

Daylight Saving Time rules for your location can also be set in the menu so the clock can deal with this nuisance automatically. The default rules are for most of North America. Automatic DST can also be disabled if your location doesn't use it. See the nine menu items starting with **Item 23 - Automatic DST** on Page 51 for these settings.

More detailed information about setup and operation can be found in the full manual!

Arduino IDE Guide

This firmware is best edited and compiled using PlatformIO. However, with a bit of doing the Arduino IDE (versions1.8 or 2.3) can be used. This has been tested on both versions of the IDE and it worked. But as they say, “Your mileage may vary.”

This guide assumes that you are familiar with the operation of the Arduino IDE.

If the code was supplied on a flash drive or other media, the files necessary are in the **EdgeLit_NTP_Clock_v4.44_Arduino** directory. There is the usual **.ino** file with the main code, and also **definitions.h** and **version.h**. All three files must be in that directory.

If you are getting the code from its GitHub repository the Using Arduino IDE file there has instructions to help you copy the files needed to a local directory.

You won't have to change anything in any of these files unless you want to modify the firmware.

First make sure the ESP32 extensions are installed in the IDE and they are up to date. Then the following libraries have to be installed. Use the IDE's **Library Manager** to search for them one by one, select the appropriate version, and click install.

Tzapu **WiFiManager** v2.0.17

Matthias Hertel **OneButton** v2.5.0

ThingPulse **ESP8266 and ESP32 driver for SSD1306 displays** v4.5.0

Sparkfun **TMP102 Breakout** v1.1.2

Daniel Garcia **FastLED** v3.6.0

Michael Margolis **Time** v1.6.1 (you may also see it credited to Paul Stoffregen)

Jack Christensen **Timezone** v1.2.4

Mitch Markin **GeoIP** v1.2.6

As of this writing, these are the current versions of the libraries. Higher versions will probably work, but if there is trouble compiling the code use the versions that were specified.

Open the **EdgeLit_NTP_Clock_v4.44_Arduino.ino** file in the Arduino IDE.

In the**Tools** menu set the following items:

Set the **Board** to **ESP 32 Dev Module**.

Set the **Port** to whichever USB port the board is plugged into.

Set **Erase All Flash Before Sketch Upload** to **Enabled** the first time the code is uploaded. This will make sure the SPIFFS file system is formatted properly so the firmware will work. It can be set to **Disabled** for uploads after the first one.

Make sure the **Partition Scheme** is set to **Default 4MB With SPIFFS**.

The rest of the default settings should be OK.

PlatformIO would have found and installed all the libraries and done all the setup by itself.

Click the **Check-mark** icon and a few minutes later hopefully the code will be compiled. If you are feeling lucky just click the **Arrow** icon and it should upload to the ESP32.

When the upload finishes the firmware should start running and you can follow the instructions in the **Quick Start First Time Setup Guide** on Page 66. The IDE's serial monitor can be opened to check that everything is working properly (see Page 60).

PlatformIO IDE Guide

PlatformIO runs as an extension for Microsoft's VS Code. Both are free to download and use without restrictions.

If you have to install PlatformIO first download and install VS Code. Then use its **Manage** function (the gear icon) to add PlatformIO. Click on **Extensions**, search for PlatformIO in the list that appears, and click **Install**. PlatformIO includes support for most Arduino boards, most ESP boards (8266 and 32), and many other microcontrollers. No further extensions or libraries need to be installed.

If the code was supplied on a flash drive or other media, all the files PlatformIO needs for this firmware are in a directory called **EdgeLit_NTP_Clock_v4.44_PIO**. If you are getting the code from its GitHub repository, it's already set up for PlatformIO. Just download the repository to a local directory.

To compile or edit the code click **File>Open Folder** in VS Code. Then navigate to where the code's directory is stored on your computer and click on it. Then click **Select Folder** and you should be good to go. The functions are in the **main.cpp** file in the **src** directory. The declarations for the variables, functions, objects, and libraries are in the **definitions.h** file in the **includes directory.**

platformio.ini is the file that lets Platform IO do its “magic” and perform all the setup that causes so many headaches with the Arduino IDE. That's where the library names and their versions are specified as well as what kind of development board or microcontroller is being used. Once all that is specified, PlatformIO automatically finds and installs everything needed to compile the code.

You won't have to change anything in any of these files unless you want to modify the firmware.

To compile the code click the **Check-mark** icon at the bottom left of the window. The A**rrow** icon next to it uploads the firmware to the ESP board. PlatformIO will even find the board automatically once it's plugged into any USB port.

When the upload finishes the firmware should start running and you can follow the instructions in the **Quick Start First Time Setup Guide** on Page 66. PlatformIO's serial monitor can be opened to check that everything is working properly (see Page 60).

That was a very quick overview. However here is a link to an excellent tutorial with a lot more information on getting started with PlatformIO and why it's better than the Arduino IDE. It shows exactly how to install it and how to define the board and libraries if you are starting a project from scratch. It also includes lots of examples.

Getting Started with PlatformIO - Better than the Arduino IDE (dronebotworkshop.com)

Here's another good one. It's shorter and not as detailed as the first one but it will get you going.

Getting Started with VS Code and PlatformIO IDE for ESP32 and ESP8266 | Random Nerd Tutorials

Happy tinkering!

-30-