OpenWordClock

Release 0.2.3

Jacob Nuernberg
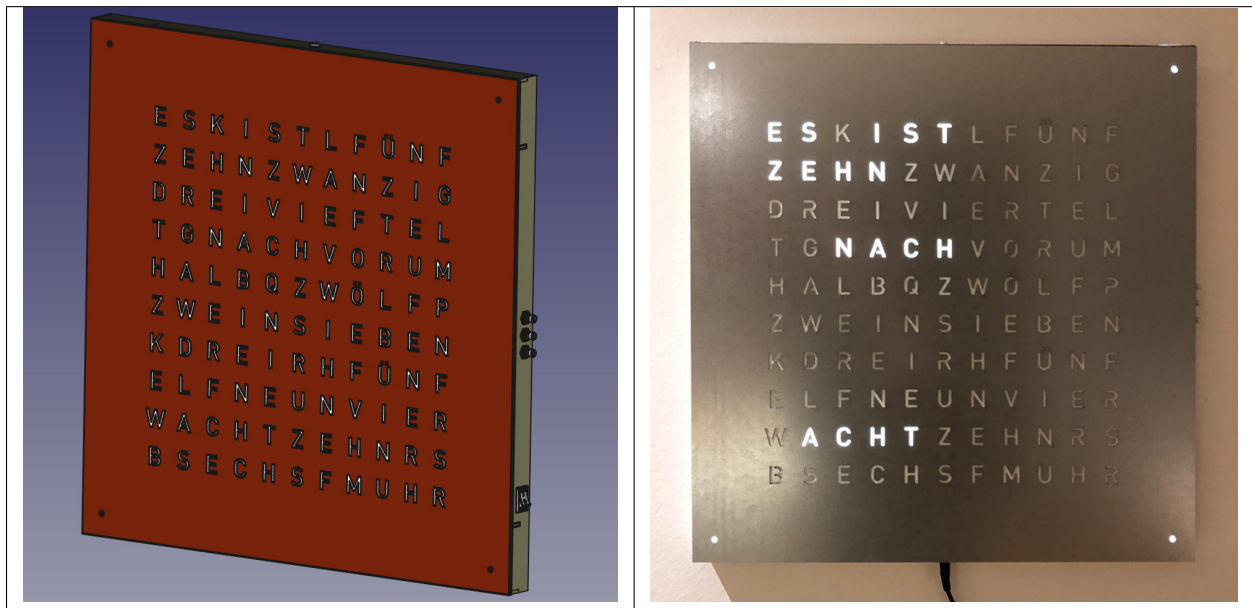
Mar 06, 2023

CONTENTS

OpenWordClock is a fully open source wordclock that can be realised with hobbyist tools and capabilities.

The project repository is hosted on GitLab.

I always fancied owning an official Qlocktwo, however found them to be rather expensive. On the other hand I was fascinated by the idea to realize one as my own project. When I then came to know about a makerspace near me with the capabilities of laser cutting and 3D-printing, I set out to realize my idea.

My aim throughout this project was to design it in such a way, that any hobbyist with a basic skill set and the will to learn new things could realize this project. The body work is laser cut or 3D-printed, the electronics are off the shelf components and the source code is build in the Arduino environment.

ONE

CONTENTS

1.1 Overview

The concept of a wordclock is widely known, and I think it does not need further introduction: It reads the time to you.

I am also not the first person to turn this clock into a DIY project. A quick web search for the terms "DIY" and "wordclock" yields plenty or results. My realisation is also not the most advanced. Nor the simplest. Or even the fastest to realize. So why did I construct my own, instead of using one of the plenty models around the interwebs?

Well, first of all, I like to tinker and fancied the idea to do it all on my own. I wanted to start on a clean slate. Had I known how much time one can spend when starting with a clean canvas, I'd probably better have chosen an existing realization.

Let me briefly describe my design paradigms before digging into the construction and how this clock works. If you are still following then, I will point you to further material in case you wish to realize your own OpenWordClock.

1.1.1 Design paradigms

From the very start of this project, I tried to follow a few but significant design paradigms for everything related to this project. In no particular order, these are.

1. OpenWordClock should as closely as possible resemble the appearance of the original Qlocktwo.

2. All parts of the project must be realizable as a hobbyist with access to a public makerspace or similar.

3. OpenWordClock was always intended to be published free and open source.

Of course, I had to adapt the face of OpenWordClock to match commercially available components. Also, the choice of microcontroller limits the number of features to a subset of the original.

To be able to realize all this as a hobbyist, I resorted to using free and open source software, whenever possible. The 3D models and technical drawings are realized in FreeCAD, further sketches are done with Inkscape. To design the logic board, I used KiCAD and for source code development I used the Arduino IDE and its libraries.

For the actual build, I consulted my local makerspace at the Dynamo in Zurich, where I had access to laser cutting, 3D-printing and foil plotting equipment.

Finally, published as a free an open source project, everybody is invited to realize their own OpenWordClock. And of course everybody is invited to adapt it to their personal needs.
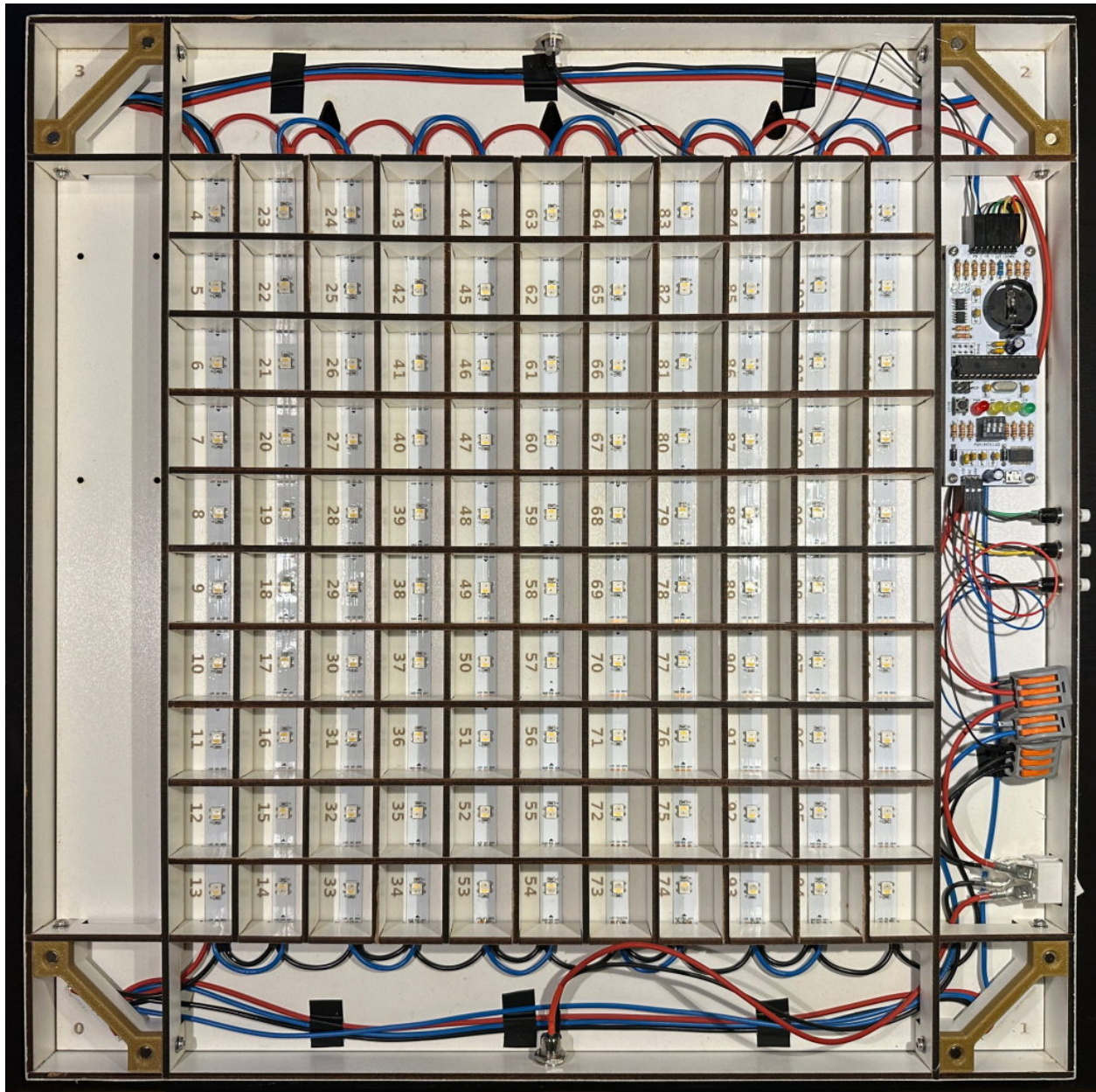
1.1.2 Construction

Fig. 1: Internal construction of OpenWordClock. The slotted structure of the separator grid and the corner blocks are clearly visible.

The body is constructed from slotted MDF parts and a matching baseplate. All is held in place with screws and some glue on 4 corner blocks with heat set threaded inserts. For a nice outer finish, the frame is wrapped with edging strips.

The outer measurement with 450 mm x 450 mm match the measurements of the original. The body is 25 mm deep (22 mm internally). The slotted structure forms a separator grid matching the pitch of commercially available LED strips with 30 LEDs per meter. The result is a square 330 mm x 330 mm grid with 11 columns of 10 LEDs. Each column contains a strip with 10 LEDs with one more LED in every corner.

The baseplate holds all electronics, apart from controls, that are placed in the outer frame. The 3 control buttons and the power switch can be either placed on the left or on the right. Similarly, the logic board can be mounted on the left

or on the right. The 5.5 mm / 2.5 mm barrel jack for power is always on the bottom. A photodiode for room brightness detection is placed in the top.

Fig. 2: OpenWordClock with laser cut zinc steel front panel. A polystyrene diffusor evens the illumination of the letters.

I tested two different styles of front panels: laser cut stencils or foil masks on diffusor plates. Both styles yield excellent results and are suitable for different finishes.

The laser cut stencil approach is best for wood or metal front panels, where one would like to preserve the surface structure of the original material. The letters are cut from the material stock, then a diffusor is applied behind all cutouts.

The foil approach uses a foil as mask, that has the advantage, that it can be custom printed unlike complex material surfaces. After printing and cutting on a foil plotter, the cutout letters are removed and the foil is transferred to an acrylic or polystyrene sheet with transfer foil.

In either case, the front panel is held on the clock with 8 magnets with matching counterparts in the corner blocks. The magnets in the corner blocks are countersunk to yield a tight fit and prevent sliding of the front panel.

1.1.3 How does it work?

Well, in essence, it's all really simple, right? RIGHT?!?

Basically, you take the time and light up the correct LEDs behind the right letters to spell out the time. But wait, there is actually a fair bit of logic running in the background.

First thing needed is a clock and while the common watch is nice and handy, it's not very suitable for such project. OpenWordClock uses a DS3231 RTC module from Analog Devices to keep track of the time. Every minute, the RTC uses an interrupt to signal to the microcontroller that another minute has passed.

The microcontroller, an ATmega328P like on an Arduino, reads out the current time and translates it to a series of indices, that correspond to the LEDs behind the right letters to light up. It then sends out a digital signal to the individually addressable WS2812B RGBW LEDs. This signal also carries color information for every LED.

Now in addition, to be able to set the time, the microcontroller also has to process inputs from connected buttons. Also for these user inputs, OpenWordClock uses interrupt based routines.

To combine these seemingly easy tasks, the source code employs a finite state machine. The state machine manages the interrupts and redirects the program flow to the right state to finally display the desired result. And using an interrupt based scheme allows for another advantage: Instead of constantly having to pull for updates, like a new time or a button press (keyword: busy waiting), the microcontroller simply has to wait for the next interrupt. This also allows OpenWordClock to put the microcontroller to sleep whenever no action is needed.

In general, the whole control logic can be put together with off-the-shelf components on a breadboard. A simple Arduino Nano paired with a DS3231 eval board connected to a few buttons and some LEDs and the electronics are up and running. In fact, this is how my first OpenWordClock was set up. For cleanliness and the sake of learning about PCB design, I created a fully integrated control board, essentially a custom Arduino Nano with a DS3231. This control board resolves a whole rats nest of wires and is neatly mounted inside the word clock.

1.1.4 Where to start?

Now, that you know the basic principles of OpenWordClock, it is up to you where to continue.

Did this introduction catch your interest and you want to build a OpenWordClock yourself? Keep reading about the *hardware and the assembly*. Find out, where you can source the components, where you can laser cut and 3D-print, and if you have the tools and the time. Still interested? Then go ahead, what's stopping you.

Are you interested in the Software that OpenWordClock employs or want to upload and test it yourself? Then read through the *source code documentation*.

Or are you already done with your build and need to know how to operate the clock? There is also an *operating manual* for OpenWordClock.

In any case, I suggest you check out the project repository hosted on GitLab where all resources are hosted. Feel free to leave remarks and report issues.

1.2 Operating manual

Please be aware, that this is a DIY project. The controls may vary in your implementation. In addition, the whole project is work in progress and the implementations may change while this operating manual is not yet fully up to date.

1.2.1 Controls

Depending on the assembly of the clock, the controls are either on the left or the right side of the clock. The controls consist of a power switch and 3 simple push buttons.

Power

The clock operates on 5 V power supplied though a standard 5.5 mm / 2.5 mm barrel jack at the bottom of the chassis. In normal operation, the clock pulls between 0.5 A and 2 A of current, depending on the selected brightness. I recommend using a power supply, that can supply at least 3 A of current at 5 V.

In addition, the clock has a power switch, that acts directly on the 5 V supply line. Switching the clock of with this switch removes power from all components, including the logic board. Therefore, all settings set by the user (e.g. color, color mode, brightness etc.) are lost, when turning off the clock. Only the time is kept due to a standby battery cell on the logic board for the real time clock module.

Buttons
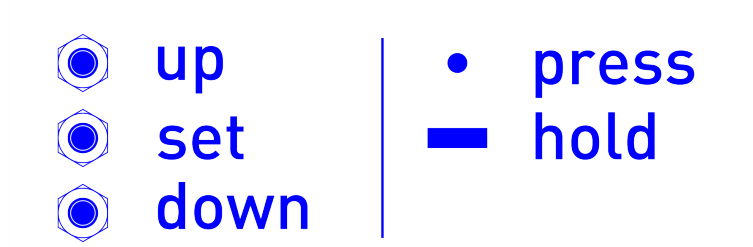
The clock features three buttons on one side. In this manual, the buttons are referred to as as up, set, and down, as depicted in the image above. All buttons support two different actions, press and hold, where a hold has to last at least one second. The notation chosen in this manual is shown on the right with for press and for hold.

1.2.2 Changing the color

When not in the menu and the clock is displaying the time, the buttons control the color of the display. The clock has three color modes and starts up with a static color.

- Pressing **set** cycles through the color modes.
  - Pressing **set** once changes the color to constant white.
  - Pressing **set** again activates color cycling mode. The display blinks once. Every time the display updates, the color is slowly advanced along the rainbow.
  - Pressing **set** again returns to constant color mode and the display will be one fix color.
- Pressing **up** or **down** returns the clock to constant color mode (if not already) and advances the color one step along the rainbow.

- Holding **up** or **down** returns the clock to constant color mode and starts cycling through all colors. Release the button at the desired color.

1.2.3 Menu

- To enter the menu, hold **set**. The display changes to image mode and displays the first menu item.
- Press **up** or **down** to navigate through the menu items.
- Press **set** to enter the according menu item
- The image below shows the five different menu items. These are
  - Set the time
  - Display the current temperature
  - Change the display brightness
  - Dynamic display brightness (experimental)
  - Configure night mode (experimental, unstable)

set time

display temperature

set brightness

dynamic brightness*

night mode*
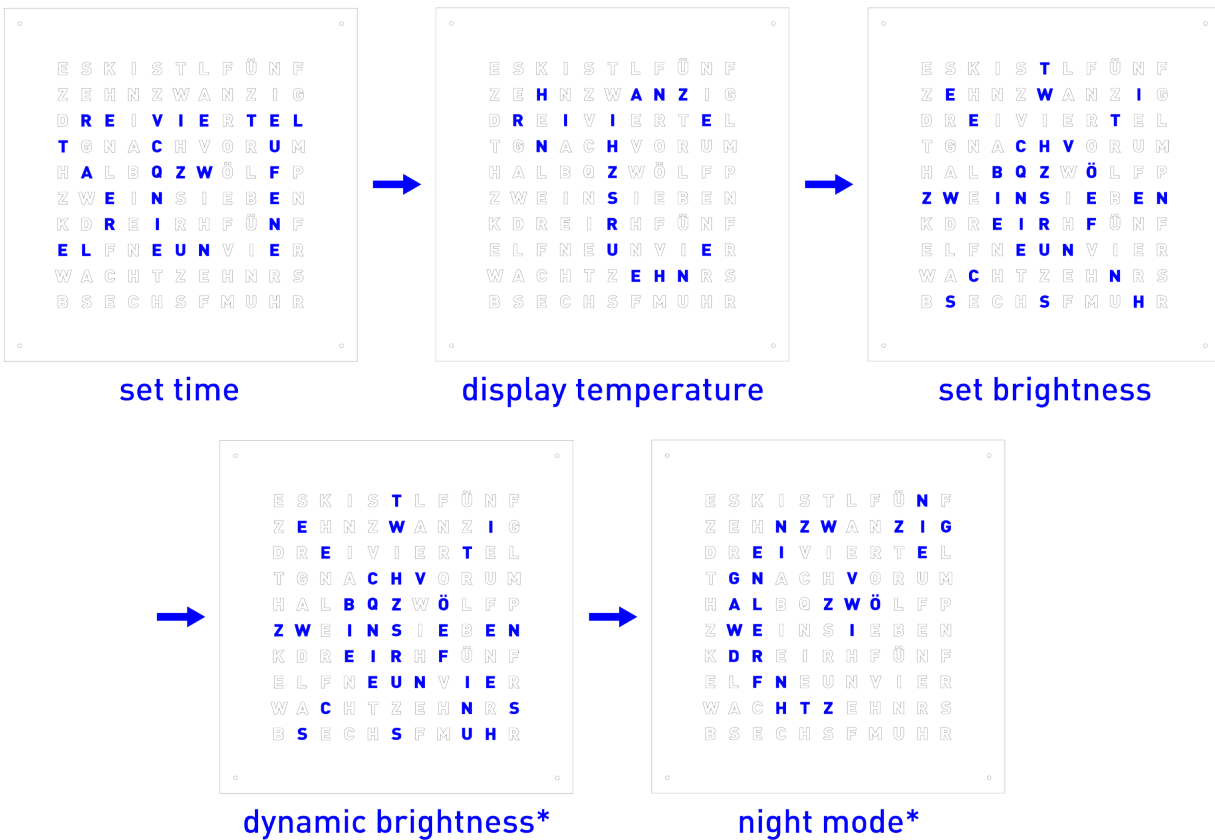
Setting the time

Navigate to the correct menu item and confirm with **set**. When setting the time, pressing **up** or **down** changes the current value and pressing **set** confirms the selection.

- The hour is blinking. Select the current hour with **up** or **down** and confirm with **set**.

- The 5 minute indicator is blinking. Select the recently passed 5 minute mark with **up** or **down** and confirm with **set**.

- The minute indicators in the corners are blinking. Press **up** or **down** to light up as many corners as necessary to complete the current minute. Confirm with **set**.

- The clock switches to image mode and displays either AM or PM. Select either with with **up** or **down** and confirm with **set**.

- Still in image mode, the clock displays the current second. Reset the second to zero with with **up** or **down** and confirm with **set**.

When completing the last step, the current time will be set.

Display temperature

Navigate to the correct menu item and confirm with **set**. The clock switches to image mode and displays the current temperature for the next minute. Press **set** to return to the time display earlier.

Changing the brightness

Navigate to the correct menu item and confirm with **set**. The clock switches back to display the time. Press or hold / **up** or / **down** to adjust the display brightness. Confirm with **set**.

Dynamic brightness

Note: This is an experimental feature. Once functionality is fully established, the manual for this feature will be added.

Work in progress.

Night mode

Note: This is an experimental feature. Once functionality is fully established, the manual for this feature will be added.

Work in progress.

1.2.4 Maintenance

Changing the battery

To keep track of the current time, even when powered of, the real time clock module has a standby battery. When removing the front panel, the battery can easily be located on the logic board. Should the battery run empty, replace it with any suitable CS2032 battery.

Board reset

Should it be necessary to reset the logic board on the clock, i.e. because an error occurred during operation, one can either switch power off, wait a few seconds, and turn it back on. Alternatively, the logic board has a reset switch on its left side to reset the program, without power cycling. Press and hold the reset button for 5 seconds. When released, the clock starts up normally.

Internal status LEDs

The logic board has 4 status LEDs soldered directly to it, that can be turned on or of with three DIP switches directly on the logic board. Depending on your choice of front panel, it is advised to turn the status LEDs of in normal operation to prevent them from shining through a not fully opaque front panel.

1.3 Hardware and assembly

Here you find all information about the hardware, where to source the individual components and how to assemble your very own OpenWordClock.

The first section goes over all parts, that are needed to build an OpenWordClock. Here, you find material information, specific part numbers and where to source them. It also covers custom laser cut parts and 3D printed components.

The following sections discuss the assembly of the OpenWordClock, starting with the body, followed by logic board and electronics with the last section being the front panel of the clock. This corresponds to the logical order of assembly. However, it is absolutely possible to work on multiple parts simultaneously, i.e. when waiting for other components to arrive.

1.3.1 Parts List

> **Warning:** The following parts lists may contain errors and are published without any warranty. Please carefully review before ordering components.

Tools

Before digging into the parts list, let's have a look at the tools needed to assemble the wordclock. Generally, these can be categorized in three categories: *definitely needed*, *access required*, and *nice to have*.

Definitely needed
 These tools are used at various points during the assembly of OpenWordClock. Make sure to have them at hand before starting the project.
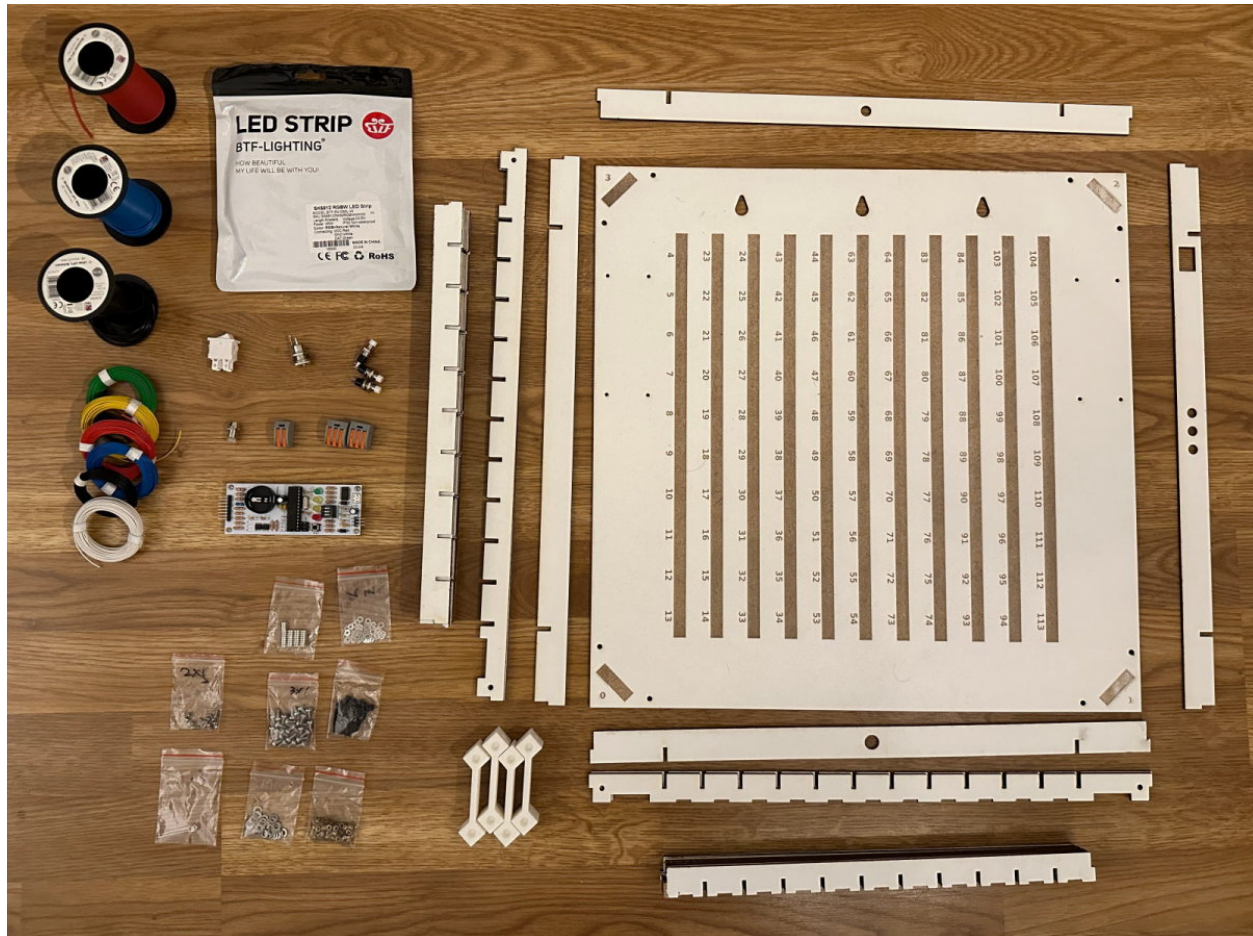
- Soldering iron

Fig. 3: Overview of all OpenWordClock parts.

- Screwdriver

Access required

These tools are needed to create the components of the body. Nonetheless, if it is not feasible to own them, you should still get access to these tools, i.e. in your local maker space or fablab.

- 3D printer

- Laser cutter

- Foil plotter

Nice to have

Albeit not strictly needed, the following tools come in handy throughout the assembly.

- Hot air SMD soldering iron

- Crimping pliers and supplies

- Tweezers

The final list contains consumables, that come in handy from time to time throughout the project.

Consumables

- Wood glue

- universal purpose glue or superglue

- solvents for cleaning (i.e. isopropanol)

Body parts and fasteners

The table below lists all parts necessary for the assembly of the body of an OpenWordClock. All laser cut parts are cut from a single 600 mm x 900 mm sheet of 3 mm thick MDF using this template. For a nice surface finish, I recommend using MDF with a white surface coating. The stock material as well as the edge band can be sourced in your local hardware store.

The corner blocks are 3D printed in PLA with 5-15 % infill and can be printed with rather thick layers (0.15 mm). As these are internal parts, they are not seen in the final outcome.

Table 1: Body Parts

Qty.	Description	Material	Comment
1	Baseplate	MDF (3mm)	Laser cut
2	Vertical long	MDF (3mm)	Laser cut
2	Horizontal long	MDF (3mm)	Laser cut
10	Vertical short	MDF (3mm)	Laser cut
9	horizontal long	MDF (3mm)	Laser cut
1	Top	MDF (3mm)	Laser cut
1	Bottom	MDF (3mm)	Laser cut
1	Left	MDF (3mm)	Laser cut
1	Right	MDF (3mm)	Laser cut
4	Corner block	PLA	3D print
1	Edge band > 25 mm wide		approx. 2 m

The body is held in place by a bunch of fasteners listed in the table below. The table also lists fasteners needed to attach the control board and the front panel. These should also be available in your local hardware store.

Note: The table lists screws explicitely as torx screws. However, any other suitable screw head fits the purpose as well.

Table 2: Fasteners

Qty.	Description
16	M3 heat set insert
16	M3x6 Button head torx screw
16	M3 Washer
4	M2x6 standoff
8	M2x5 Button head torx screw
8	M2 washer
16	Magnet round 5 mm x 2 mm

For the front panel, the required materials and parts depend on the preferred style of front panel. The table lists the parts for the two styles of front panel, that I have worked with.

Table 3: Front panel components

Qty.	Description	Dimensions
1	Polystyrene sheet (3mm)	450 mm x 450 mm
1	Foil stencil	ca. 500 mm x 500 mm
or		
1	Laser cut stencil in material of choice	450 mm x 450 mm
1	Polystyrene diffusor (0.5 mm)	approx. 0.2 m^2

Control board and electronics

The control board of the OpenWordClock can be realized in two ways: Either you take an off-the-shelf Arduino Nano and pair it with a DS3231 RTC evaluation board or you decide to order and assemble the custom control pcb according to the schematics provided in the repository. For the custom PCB, consult your preferred PCB producer; usually it is possible to order a small quantity of boards. The table below lists the components needed for the latter case. The part numbers refer to the part on Digi-Key, where I sourced all components.

Note: The tables list the quantities needed for a single control board. However, ordering in bulk may significantly reduce the price for the individual board. That is true for the board on the one hand side, and for the components on the other side as well.

From my experience, ordering 10, instead of 1 board, may cut the price per board in half.

Table 4: Custom control pcb comonents

Qty.	Description	Value	Digi-Key part No.
1	Custom PCB		
1	Battery holder BS-7		BS-7-ND
1	Battery CR2032		2984-CR2032-H-ND
3	Capacitor	1 uF	445-FK24X7R1H105KR000-ND
5	Capacitor	100 nF	399-9877-1-ND
2	Diode 1N5818		2368-1N5818-ND
1	Micro-USB connector		WM1714CT-ND
1	Pin header 90° 1x3		WM19494-ND
1	Pin header 90° 1x8		WM4106-ND
1	Pin header 2x3		WM23646-ND
1	Pin header 2x4		WM8140-ND
1	LED	Green	732-5017-ND
2	LED	Yellow	732-5018-ND
1	LED	Red	732-5016-ND
4	Resistor	10 k	10.0KXBK-ND
4	Resistor	1 k	1.00KXBK-ND
4	Resistor	4.7 k	4.7KQBK-ND
4	Resistor	1 M	1.0MQBK-ND
1	Thermal resettable fuse MFFSMF050		MF-FSMF050X-2
1	Miniature push button		450-1650-ND
1	DIP switch 3x		208-3
1	USB controller FT232RL		768-1007
1	Microcontroller ATmega328 p		ATMEGA328P-PU
1	DIP socket 2x14		ED3050-5-ND
1	RTC DS3231MZ		DS3231MZ+
1	Memory AT24CS32		AT24D32D-SSHM-B
1	Crystal oscillator	16 MHz	2151-AS-16.000-18-ND

Independent of how you realize the control board, the following electronic components are needed. These are the buttons to control the clock, the LEDs and any other components to guarantee functionality.

Table 5: Other electronic components

Qty.	Description		Digi-Key part No.
1	5.5/2.5 mm low voltage barrel plug		
1	Power switch		
3	Push button		
1	5 mm LED socket		
1	Photodiode (5 mm LED packaging)		
1	LED strip WS2812B (30/m)	5 m	
2	Wago clamp 3x		
1	Wago clamp 2x		
1	Wire red 0.5 mm^2	ca. 2.5 m	
1	Wire black 0.5 mm^2	ca. 2.5 m	
1	Wire blue 0.25 mm^2	ca. 2.5 m	
1	Signal wire various colors 0.14 mm^2	ca. 2.5 m	

Here, the red and black cable cary power to the LEDs, the thinner blue carries the signal. Various color signal wire is used to connect buttons and the photodiode. The WAGO clamps help to connect the LEDs.

1.3.2 Body

The figure below shows all parts and fasteners needed for the assembly of an OpenWordClock. Note, that some of the MDF body components are stacked up upon each other. To identify the individual parts in this picture, please refer to the *body parts lists*.

While of-the-shelf components are used for most part of OpenWordClock, the body is make up of many custom design parts. So before starting with the assembly, the custom parts have to be manufactured.

Custom parts

The body consists of custom designed slotted components from 3 mm thick MDF. These components are designed to be laser cut from a single sheet of 900 mm x 600 mm stock material. These measurements are supported by most commercially available laser cutters. As stock material, one can either use uncoated MDF, however using white painted MDF results in a nice internal and external surface finish and can usually be purchased at your local hardware store. Make sure to buy the variant with both sides painted.

For laser cutting, figure out, if you can find a local makerspace of fablab with a suitable laser cutting tool near you. Usually, you can go there and laser cut your stock material yourself in exchange for a small support for the makerspace. Also consult with the local staff, more often than not, they are able and willing to provide valuable support on their tool, such as on cutting parameters and material dependent settings.

Once cut, clean the components with a little soap water and remove any laser cutting residue or burn marks. Make sure not to soak the components (MFD will swell and deform when in contact with water) and let them properly dry afterwards.

For the 3D-printed corner blocks, you an also consult with your local makerspace. It is rather common to find a 3D-printer there. Another possibility would be a colleague or friend in possession of a printer. Or you may even own one yourself.

I printed my parts in PLA, however, any other thermoplastic should work as well. Make sure though to use a reformable thermoplastic, as after the print, heat set threaded inserts will be installed to allow for mounting.

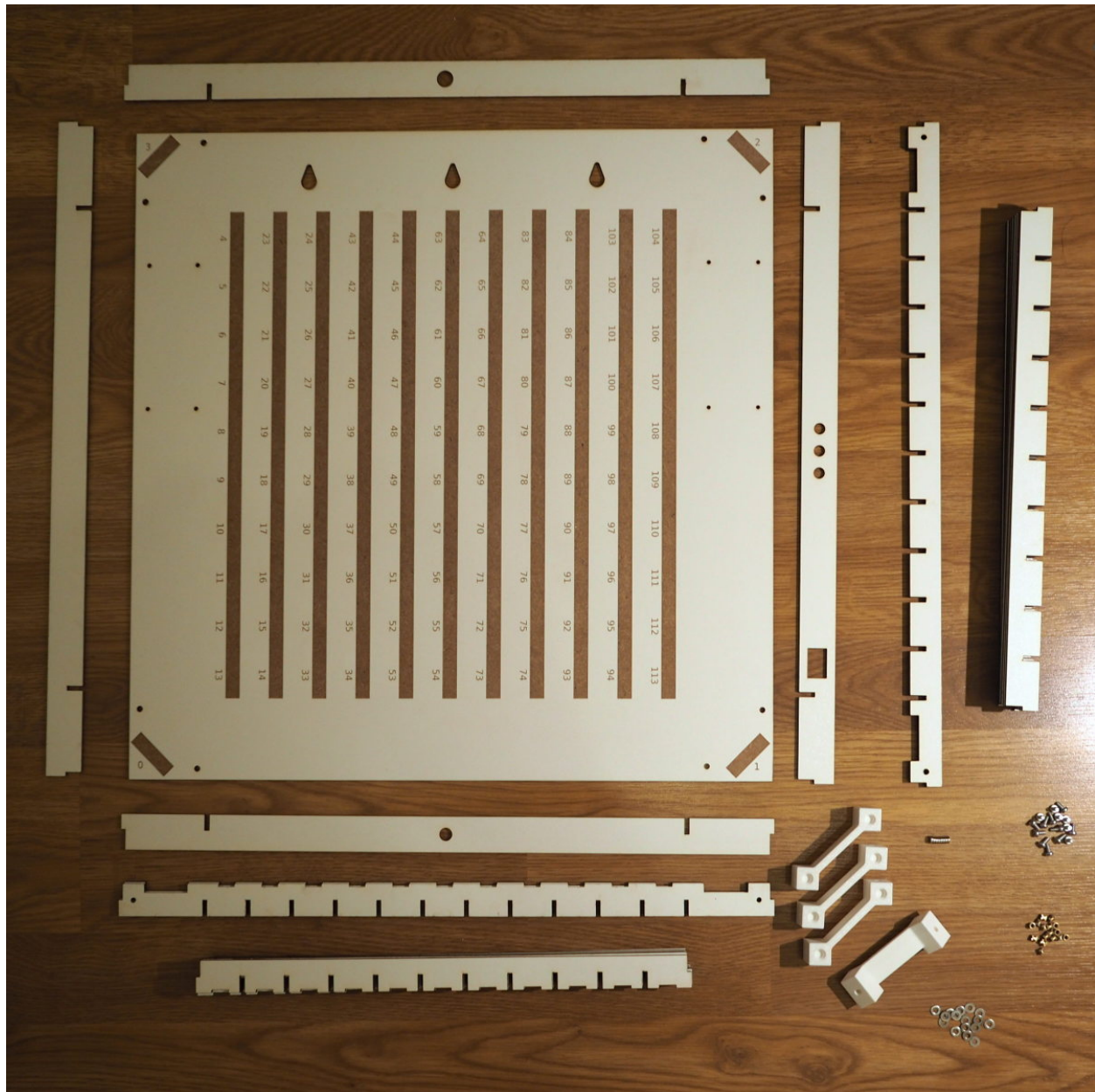Now, that you have all custom parts, it's time to start the assembly of the body.

Fig. 4: All body parts and fasteners needed for the assembly of the body of an OpenWordClock.

Assembly

The following step-by-step instructions will guide you through the assembly of an OpenWordClock.

1. Start out by inserting four threaded M3 head set inserts into the associated holes in the 4 corner blocks.

    - **Parts needed**:

        – 4 corner blocks

        – 16 M3 threaded heat set inserts

    - Pay attention to where you install the inserts, as the corner blocks contain 2 additional holes for ambiguity. Refer to the images below for the right position.

    - The heat set inserts are easily installed with the tip of a soldering iron. Reduce the temperature to avoid burning or scorching the 3D print.

    - Install one magnet in each of the counter sunk holes on the top of the corner blocks with universal or super glue.
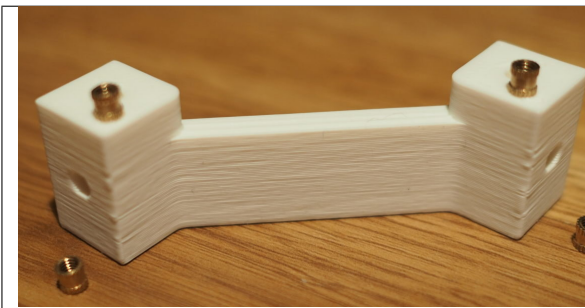

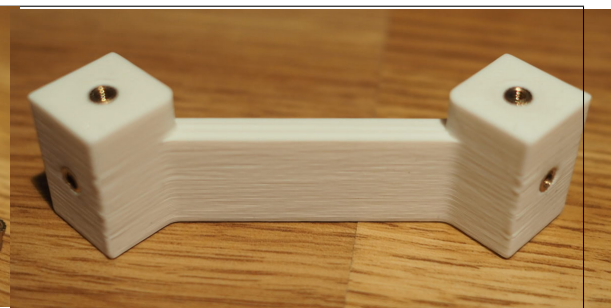Fig. 5: Corner block with M3 threaded heat set inserts before assemlby.

Fig. 6: Corner block readily assembled with heat set inserts.

2. Assemble the grid structure.

    - **Parts needed**:

        – 4 corner blocks with heat set inserts

        – 10 body parts vertical short

        – 9 body parts horizontal short

        – 2 body parts vertical long

        – 2 body parts horizontal long

        – 8 M3 washers

        – 8 M3x6 machine screws

    - Start assembling the grid structure by slotting in the 10 vertical short and 9 horizontal short body parts.

    - Install the vertical long and vertical short body parts along the outside of the assembled grid.

    - In each corner install a corner block with 2 M3 screws and matching washers through the holes in the vertical long and vertical short body parts.

        – Pay attention to have the slotted cable feed through on the same side as the feed through cut outs for the LEDs in the horizontal short and long body parts.
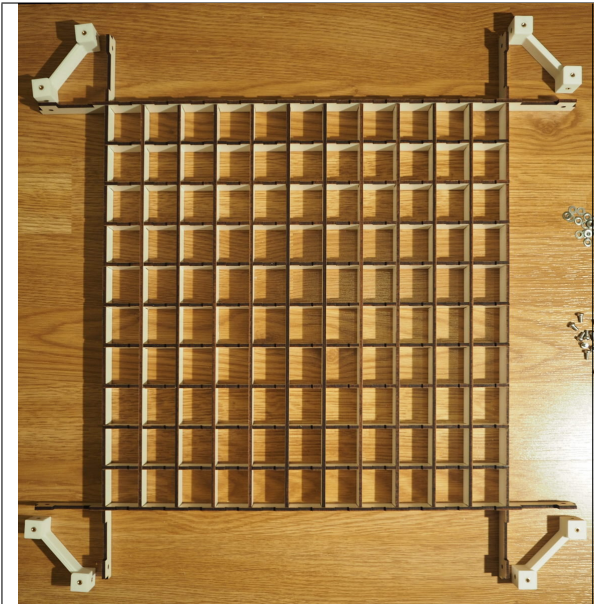
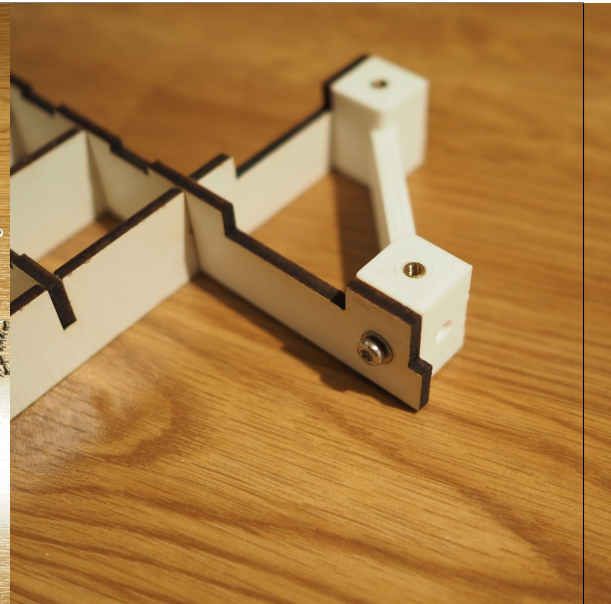Fig. 7: Grid structure with corner blocks.

Fig. 8: Corner block installed on grid structure.

3. Add the base plate.

- **Parts needed**:

    - 1 baseplate

    - 8 M3 washers

    - 8 M3x6 machine screws

- Install the base plate with 8 M3x6 machine screws and matching washers from the back.

- Pay attention to having the engraved position marks for the LED stripy running vertically through the grid.

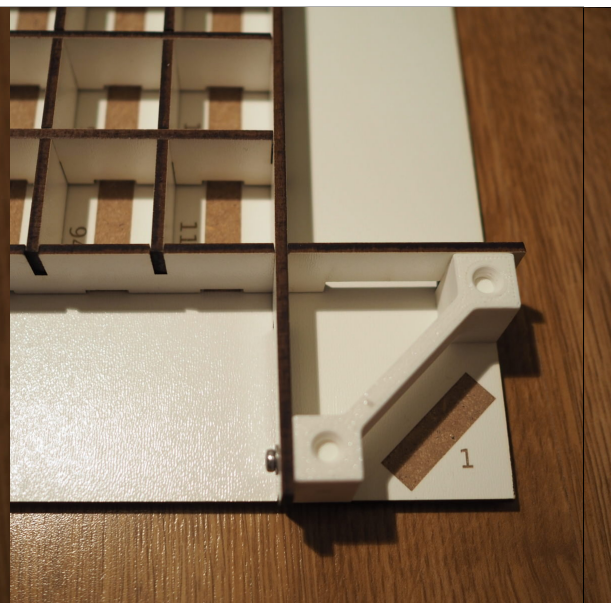
Fig. 9: Base plate fixture points.

Fig. 10: Corner with base plate in position.

4. Add the frame.

   - **Parts needed**:

     - 1 body part top

     - 1 body part right

     - 1 body part bottom

     - 1 body part left

     - Wood glue

   - At this point, you have to decide whether your OpenWordClock shall have the controls on the left or on the right hand side. Depending on your decision, you have to place the body frame part accordingly. Identify where each frame part goes.

     - The top frame part has a small centered hole for the photodiode.

     - The bottom frame part has a larger centered hole for the power supply barrel socket.

     - The part with 3 holes for buttons and one rectangular cutout for the power switch houses the controls. You decide, whether this is on the left or on the right.

     - The last part has no cutouts and goes either on the left or right, depending on your decision above.

   - Apply some wood glue and everywhere, where the frame parts come in contact with the grid, but not the base plate. Also apply wood glue in the corners, where the frame pieces touch.

   - Align, position, and fix in place the frame pieces. Some painters or electrical tape around the corners helps to keep the pieces in place while curing (not yet applied in images).

> **Warning:** Make sure, that you don't glue the frame to the base plate. You will want to remove the base plate later to install the electronics.

Fig. 11: Wood glue applied on corner block.   Fig. 12: Frame installed and positioned for curing.

5. Apply edge band

- **Parts needed**:

    - Edge band in color of choice, approx. 2 m

- Once the glue has cured, apply edge band all the way around the frame.

    - The cutouts will later be cut free of the edge band again.

    - Word carefully around the corners to prevent ugly breaks in the edge band.

    - Place the start and end of the edge band on one of the upper corners, as they are like less visible when wall mounting the OpenWordClock.

- With a sharp carpenters knife, cut of excess edge band.

- If necessary, sand down the edge band edges for a good finish.

Congratulations, you have successfully assembled the body of the OpenWordClock. When ready, continue with the *logic board and control electronics*.

1.3.3 Logic board

Generally, there are two approaches to setting up the microcontroller and clock unit inside the clock.

The first option consists in pairing an Arduino Nano with a DS3231 RTC module. Connections are realized with simple jumper wires, alleviating the need to solder a lot of connections. However, this approach adds a few wires to the assembly and there is no dedicated mounting point inside the OpenWordClock to mount Arduino and RTC.

To reduce wire cluttering and cleaning up the internals, I have designed a custom circuit board, that combines Arduino and RTC module on a single PCB. The PCB features dedicated connectors for buttons, LEDs etc.

Option 1: Arduino and RTC

For this approach, there is no need to solder any components to a PCB and additionally, the components can be acquired as off-the-shelf evaluation boards: All needed are an Arduino Nano and a DS3231 evaluation board. In fact, the entire hardware and source code development was done on such a setup. The image below depicts the necessary connections.

Note: If feasible, you may also use a breadboard for the assembly. However, it is not guaranteed, that the breadboard will fit into the OpenWordClock.

- Connect the clock module to the Arduno Nano

    - Connect the VCC pin to the 3.3 V pin on the Arduino Nano

    - Connect the GND pin to any Arduino Nano GND pin

    - Connect the SDA and SCL lines to the SDA (pin A4) and SCL (pin A5) pins on the Arduino

    - Finally, connect the SQW pin to pin 2 (D2) on the Arduino to receive interrupts of the RTC.

- Prepare the photodiode with a 1 MOhm resistor between the leads. Make sure to also attach wires to both leads and leave enough spare for later connections.

For now, that's it. Check the image above for all other connections, that have to be made later on.

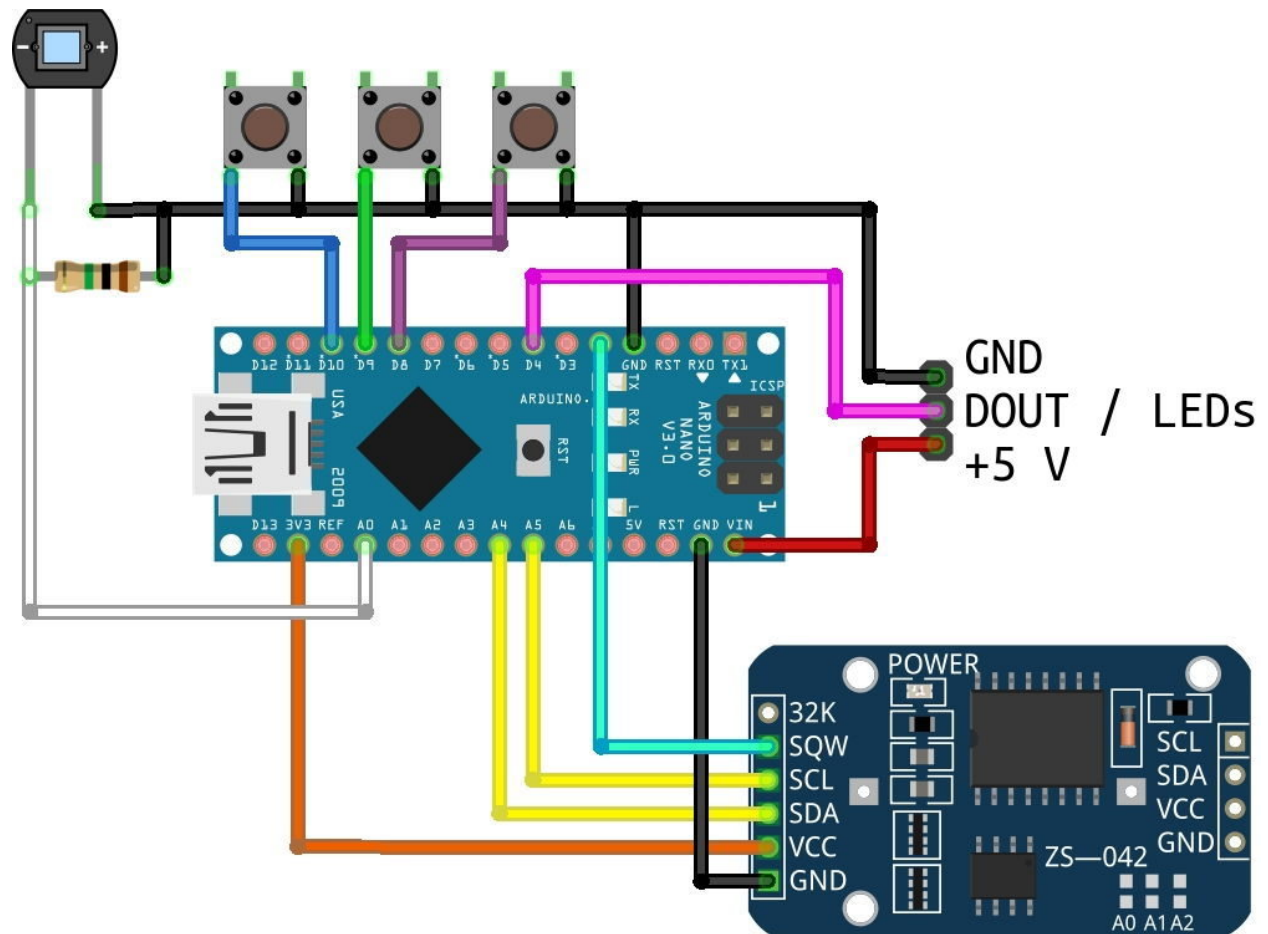Fig. 13: Edge band applied before cutting off excess.

Fig. 14: Evaluation board approach with all connections needed for the OpenWordClock. The connector on the left hand side is indicating the connections for power supply and data communication towards the LEDs.

Option 2: Custom PCB

To remove wire clutter and integrate all electronics on one logic board, I designed a custom PCB, that combines the Aruino and the RTC module. In essence, it is still a simple Arduino Nano with the RTC module on the same PCB and matching connections for the OpenWordClock. The image below shows the PCB in raw and assembled state.

After acquiring the PCBs from your preferred PCB manufacturer, populate it according to the list, that can be found on the repository. While most of the components are through-hole (THT) components, I couldn't find suitable THT versions of some. These remaining components are SMD components. There are several different techniques to solder SMD components with a common one being a hot air soldering iron.

Once fully assembled, the logic board has all components and connectors, that are needed for the OpenWordClock.

1.3.4 Electronics

Now, that the body is ready, it is time to equip it with the electronics. At this point, a lot of soldering will come up, so be aware and have some time on your hands. It is also recommended to word properly to avoid and shorts or to break any components.

Installing the LEDs

In a fist step, it is time to install the LEDs on the base plate.

1. Remove the baseplate from the frame again and put the frame aside.

2. From the WS2812B RGBW LED strip, cut of 11 sections of 10 LEDs each.

3. Glue the LED strips to the engraved positions on the base plate.

Important: Pay attention to have the data in and data out directions aligned properly. Take the LED index numbers as help to align data in and data out properly. You should end up with alternating directions for the LED strips.

---

4. Cut of another four individual LEDs.

5. Glue the 4 individual LEDs on the engraved marks in the corners.

   - Also pay attention here, to have data in and data out aligned with the index numbers.

   - The image below shows the base plate after installation of the LEDs.

6. Wire up the ground (GND) terminal to the LEDs

   - Use black wire with 0.5 mm^2 cross section

   - On the top side, make hops from one strip to the next to supply the GND line to every strip

   - Make sure to leave about 50 cm of spare wire at the first LED.

   - From one of the top corners (index 4, or 104), connect to the nearest corner (index 3, or 2), then cross over to the other top corner.

   - Similarly connect the bottom corners (index 0, or 1) to the bottom corners of the grid (index 13 or 113).

7. Wire up the +5 V line to the LEDs

   - Use red wire with 0.5 mm^2 cross section

   - Distribute the power in a similar manner as the GND line, just along the bottom of the grid.
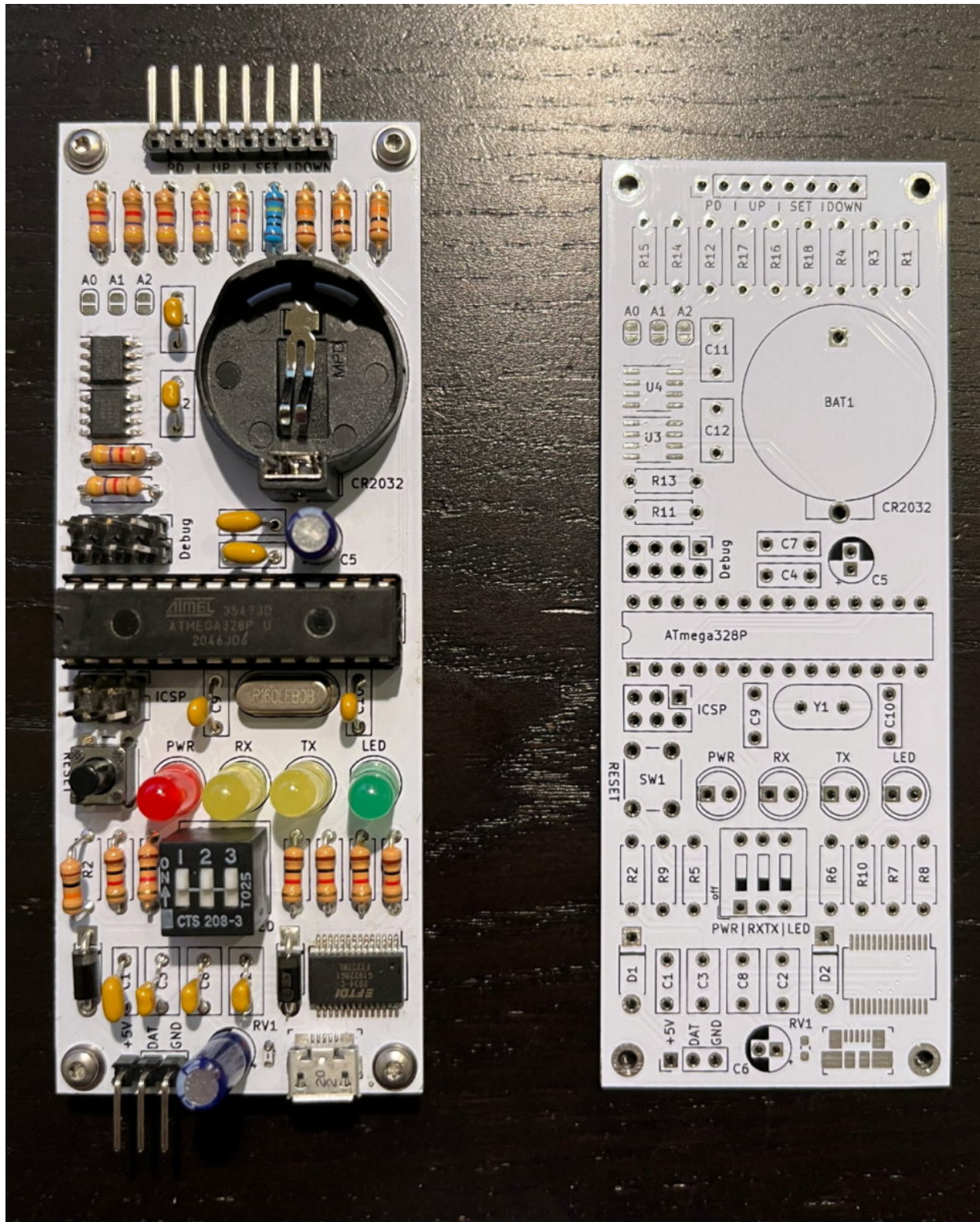

Fig. 15: Custom OpenWordClock PCB in assembled and raw state.

- Again make sure to have about 50 cm spare wire at the first injection point.

- In an equal manner, wire up the corner LEDs.

Note: In the image below, +5 V (top, red) and GND (black, bottom) are inverted to these instructions. Following the instructions leads to the data line always being on the inside of a GND or +5 V wire loop.

---

8. Solder in the data line.

    - Use blue signaling wire to distribute the data.

    - Start from the highest index and follow the data line in inverse order. Everywhere, where there is a connection needed connect to the next lowest index.

    - You should end up with alternating loops on the top or bottom of the grid.

    - Connect the corners in the same manner.

    - Make sure to have approx. 50 cm spare signal wire at the data in (DIN) terminal of the LED with index 0.

9. Use some electrical tape to fixture the loose wire bundles going around the outside of the base plate.

10. To finish up, reinstall the baseplate in the frame. Carefully feed all wires through the wire feed through cutouts in the corner blocks and vertical and horizontal long pieces.

The image below shows the wired up LEDs. Note, that positions of GND (black, bottom) and power (red, top) are inverted with respect to these instructions.
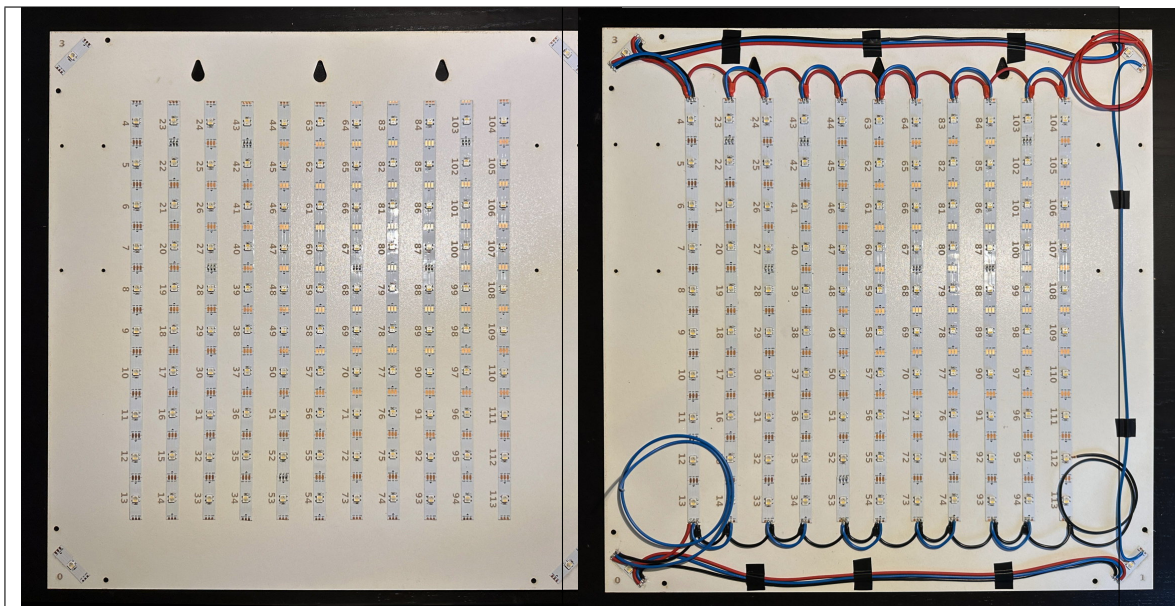

Fig. 16: WS2812B LEDs installed on base plate. Note the alternating direction of the LED strips.

Fig. 17: Wired up LEDs with +5 V (red), GND (black), and data (blue).

Installing controls and logic board

1. If not done already, cut openings into the edge band, where there are cutouts for buttons, power switch, barrel socket and photodiode in the frame.

2. Prepare the barrel socket, power switch, barell socket and photodiode with wires of suitable length like in the image below.

   - Use at least 0.5 mm^2 wire for +5 V and GND lines.

   - For buttons and photodiode as well as for power supply to the logic board (not the LEDs) 0.14 mm^2 signal wire suffices.

   - Note, that the image shows crimped connections in various points, however soldered connections work just as well.

3. Install the barrel socket in the bottom of the frame.

4. Install the power switch and the three buttons on the right side of the frame.

5. Install the photodiode in the top of the frame.

6. Install the logic board with 4 M2x8 mm standoffs and 8 matching washers and M2x5 machine screws. Note, that you can run the wires for the buttons underneath the logic board.

7. Connect photodiode and buttons to the top connectors of the logic board.

8. Connect the barrel plug to the power switch. From there run each one ground and +5 V connection to a 3 power wago clamp.

9. Connect the GND and +5 V lines from the LEDs to the same Wago clamps.

10. Connect three wires to the GND, DOUT, and +5 V connectors at the bottom of the logic board.

11. Attach the GND and +5 V wires to the Wago clamps from before.

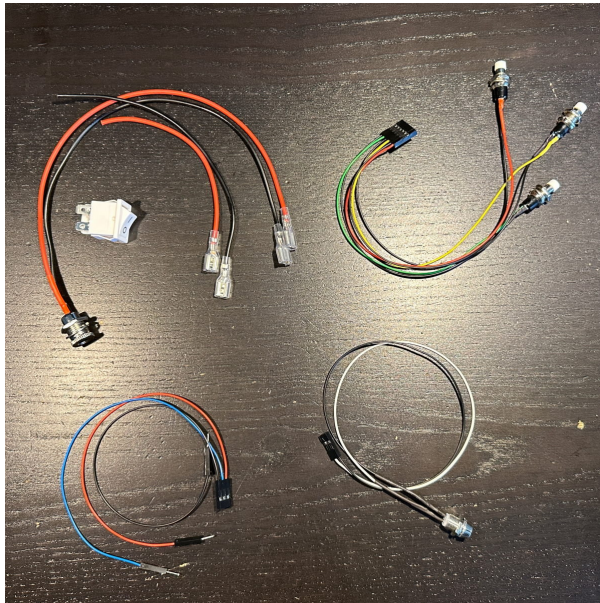12. Finally, use a 2 port Wago clamp to wire out DOUT of the logic board to the data line of the LEDs.

Fig. 18: Prepared wires and components for power (top left), control buttons (top right), control board supply (bottom left), and photodiode.
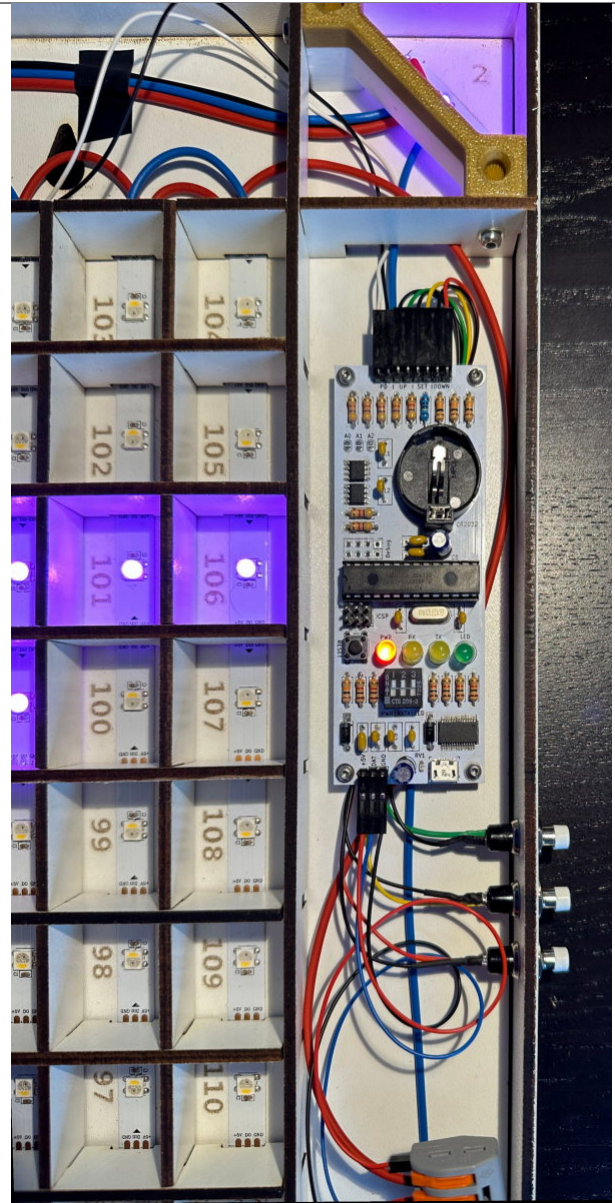

Fig. 19: A closer look at the logic board with connections for buttons and photodiode at the top and GND, DOUT, and +5 V at the bottom.

If everything went right, you are now ready for a first function test. Load up a suitable test script to the logic board and supply power to the OpenWordClock. Hopefully, you see some LEDs lighting up. The image below gives you an overview over a fully wired up OpenWordClock.
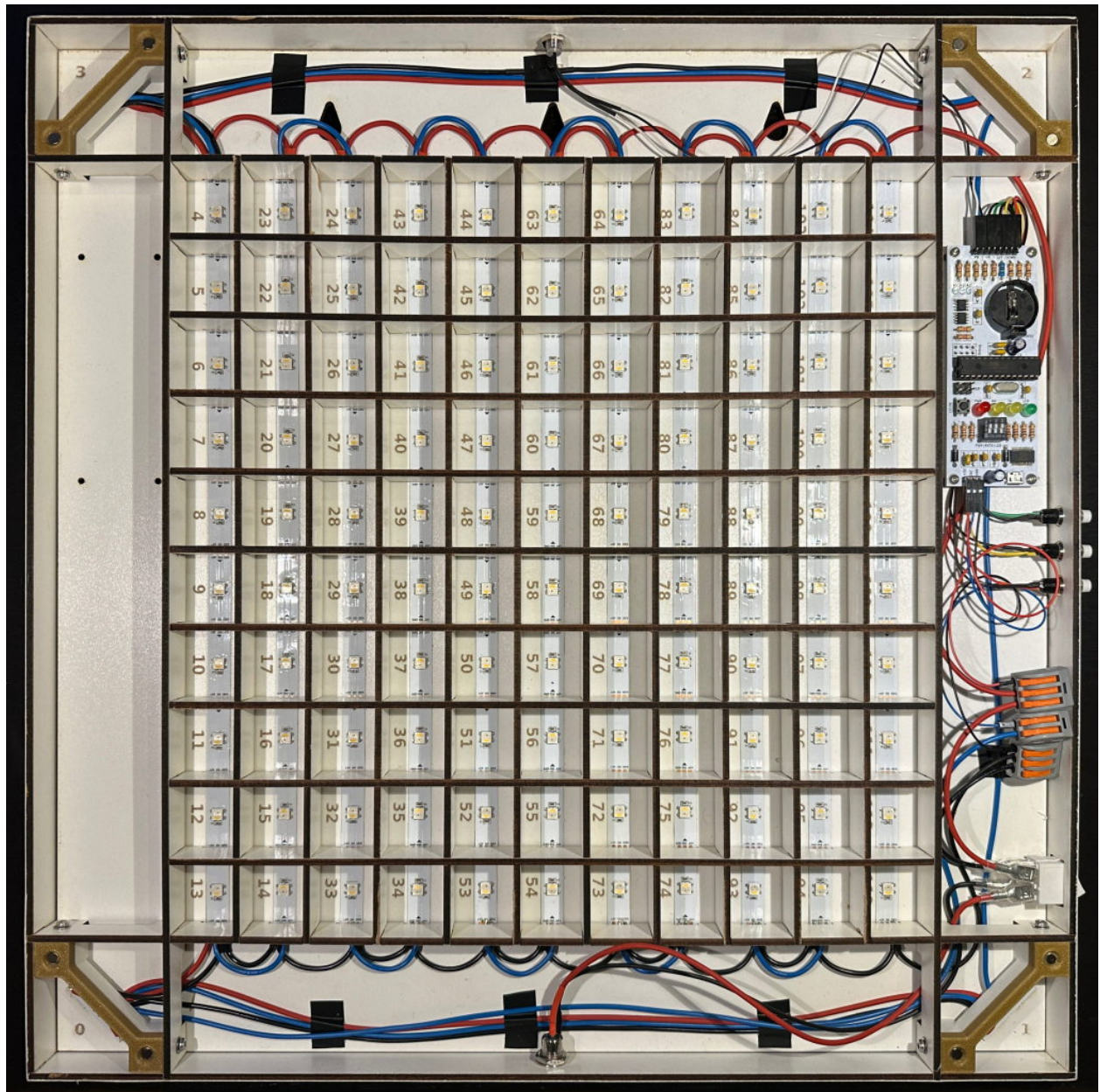
Fig. 20: Internals of an OpenWordClock with all connections made.

1.3.5 Front panel

Finally, it is time to give OpenWordClock its face. Within this project, two different styles of front panel are discussed. However, there are more techniques, that can equally be used to create a suitable front panel.

Front panel layout

There are some considerations to take into account, when designing the front panel. Obviously, the desired language changes the front panel design (and the software programming). But even within one language, there are certain peculiarities.

Taking the german language as an example, think about how a time is said in the spoken language. In German, to tell the time (accurate to 5 minutes), one first tells the 5-minute-mark, followed by a preposition (before, after) and then the hour. All that is preceded by "Es ist" (i.e. it is) and followed by "Uhr". We thus can construct any time to tell by following through the following list.

1. "Es ist" - always preceeds the time to tell.

2. The 5-minute mark. Select the most suitable from

    - "fünf"

    - "zehn"

    - "viertel"

    - "zwanzig"

3. The correct preposition

    - "nach"

    - "vor"

4. If applicable, the half hour - "halb"

5. Finally, the hour - "eins" … "zwölf"

We want to have these building block represented in the front panel and that in the right order. "Es ist" needs to go on the top, "Uhr" goes at the end. The 5-minute-mark needs to go before the hour.

In some german dialects, there is the weird way to tell the time in quarters of the hour, resulting in (admittedly, even for a german strangely weirds sounding) phrases like "Es ist dreiviertel vier" ("It is threequarters four", 15:45). To accommodate this variation as well, we also want to prepend "viertel" with "drei", allowing us to light up the right LEDs use either dialect.

The image above shows the rule set implemented in a front panel design. The red letters need to be positioned as shown. Obviously, there is a little variation possible (i.e. moving "nach" and "vor" to the left or right).

The green positions marked with "X" can be filled with random letters. Depending on your personal taste, you can also use images, emoticons or *bananas for scale*. Before heading to the plotter or laser, replace the "X" by letters of your choice.

Finally, to get the LEDs in the right positions, all letters need to be placed on a grid with 33 mm in the vertical direction (matching the pitch of the LED stripes). The pitch in the horizontal direction is 30 mm.

Fig. 21: A possible letter placement on the front panel for the OpenWordClock. The red letters need to be positioned as shown and allow only for very little variation. The green "X" are filled with random letters.

Diffusor materials

To get evenly lit letters, either through a foil stencil or through a laser cut front panel, OpenWordClock needs a suitable diffusor. While in the first approach, the diffusor is the front panel carrying material at the same time, only a thin diffusing layer is needed in the second approach.

Good results can be achieved with milky acrylic sheets. However, I found polystyrene to create an even better and homogeneous distribution of light. In addition, polystyrene is available either as 0.5 mm thick sheets or as 2 mm thick plates, so it can be used for either style of front panel.

The 0.5 mm thick sheets are usually available in an artists supply shop and can be cut to shape with scissors or a sharp knife. Here, no minimum size is required, but for laser cut front panels, the diffusor can consist of multiple pieces. Thicker polystyrene can be sourced in a hardware store, where it is common to order custom sizes as well. For foil front panels, the diffusor should be cut to 450 x 450 mm and be between 2 and 3 mm thick.

Foil on diffusor

The first approach consists in applying a foil stencil to a milky screen as front panel. This technique is particularly suited for custom printed front panels and glossy finishes without resorting to paint or lacquer. In contrast to the other approach, the foil on acrylic approach allows for full letters without any bridges or supports.

- Cut the front panel on a large scale foil plotter.
  - Consult your local fablab or makerspace. You may likely find a suitable foil plotter there. You need a plotting width of at least 500 mm.
  - Some plotters also support printing, so you can print your very personal OpenWordClock design.
- carefully remove all letters from the stencil
  - Suitable tools for this task are a sharp exacto knife, fine tweezers and possibly a needle.
  - Be careful not to remove the internal parts of letters such as A, D, O, or R.
- Apply transfer foil
  - Ask at your local fablab or makerspace, where you may find a suitable transfer foil.
  - Make sure to evenly apply pressure to have all parts attached.
- Remove the protective layer and apply the stencil to the diffusor
  - When removing the protection layer, make sure all internal letter parts stay attached to the transfer foil.
  - When transferring the stencil over to the diffusor, use plenty of soap water, to be able to readjust the position. Then push the water out from the center to the edges to avoid bubbles.
- Fold over the stencil edges over the diffusor to get clean edges.
- Finally, remove the transfer foil. To see how to attach the magnets, keep reading at the bottom of this page.

Fig. 22: Foil on acrylic front panel.

Laser cut front panel

This second approach of a front panel is very suitable for front panels of wood, metal or other materials with a particular surface finish. The images below show a front panel from 3 mm teak plywood.
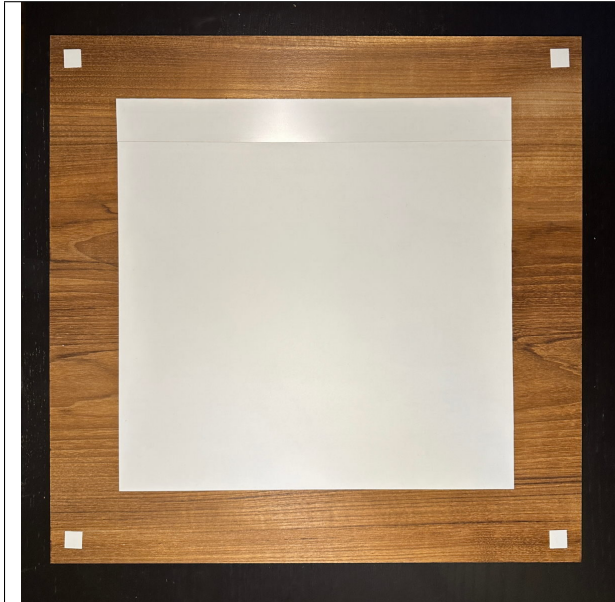

Fig. 23: Diffusor on the back side of a front panel.

Fig. 24: Final assembled front panel.

- Take your stock material to your nearest cutting laser and cut out the stencil.
    - Make sure to choose the template with the bridges in letters like A, B, O, or R to avoid ugly holes.
- Take your diffusor, measure and cut it into correctly sized pieces.
- Glue the diffusor behind the letters, either with doubly sided tape or a universal purpose glue.

That's it, done. For installing the mounting magnets, keep on reading below.

Attaching magnets for mounting

As a final step, it is time to install the mounting magnets on the front panel. There is a rather simple but powerful trick to achieve a good result.

- Take out the readily assembled body of your OpenwordClock and apply a strip of plastic wrap across the corner blocks with the countersunk magnets.
- Take the magnets for the front panel and place them on the magnets in the corner blocks. The align perfectly on their own
    - If the countersink in the corner block is slightly too deep, place an additional magnet below the plastic wrap to raise the magnet for the front panel. You can remove it later.
- Apply universal glue to all magnets.
- Place and carefully align the front panel on top. Wait until the glue has cured.
- Remove the plastic wrap and eventual additionally placed magnets. Reinstall the front panel.

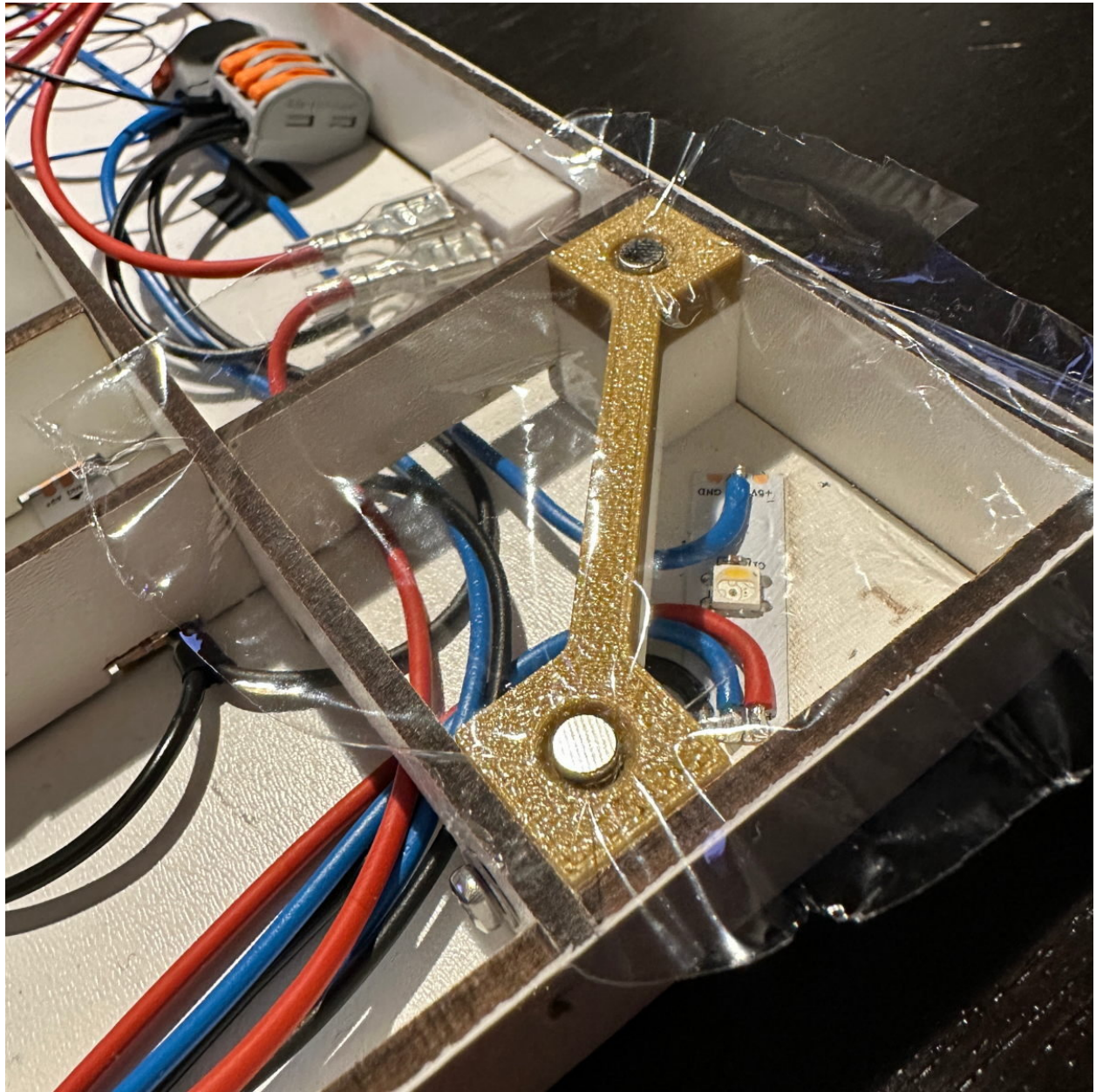Congratulations. Your OpenWordClock is now fully assembled.

Fig. 25: OpenWordClock front panel magnet installation with plastic wrap around the corners. Glue not yet applied.

1.4 Software

* Work in progress.*