

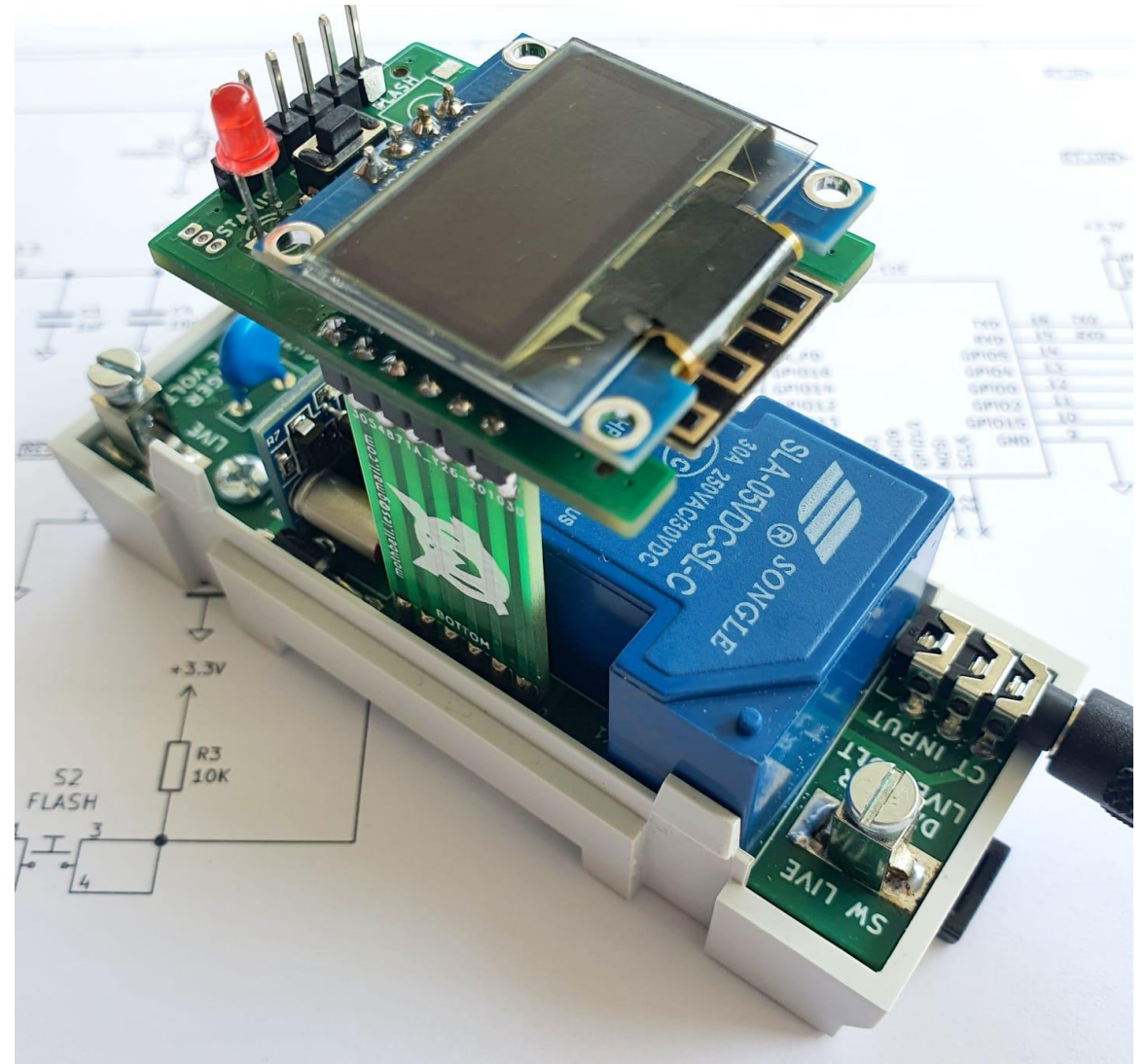
TIME SCHEDULAR / POWER ENERGY MONITOR

UNDER THE HOOD

The hardware took several design iterations to baseline. There was also a significant time spent on writing code and error checking to try and make the code “bullet proof”.

Initially the idea was just to switch the geyser ON/OFF using a scheduler but as the project developed I thought it would be neat to added a power monitoring circuit as well, this added a whole new dimension of what could be done..

The design is based on the ESP-12F WiFi platform and using the Blynk App to manage your schedule and actively monitor your power consumption. The design uses a 32mm wide Din Rail Housing that can be easily wired up into your Distribution Board.



TECHNICAL SPECIFICATION

- Operating Voltage 110 - 230VAC 50Hz
- Isolation AC-DC converter PSU
- Relay contact rating 30Amp
- Power consumption accuracy within 5% (Subject to calibration accuracy and using a fixed 230V in the calculations)
- WiFi Connectivity
- F-RAM (Nonvolatile Ferroelectric RAM) for data storage (100 Trillion Read/Write Cycles)

USES

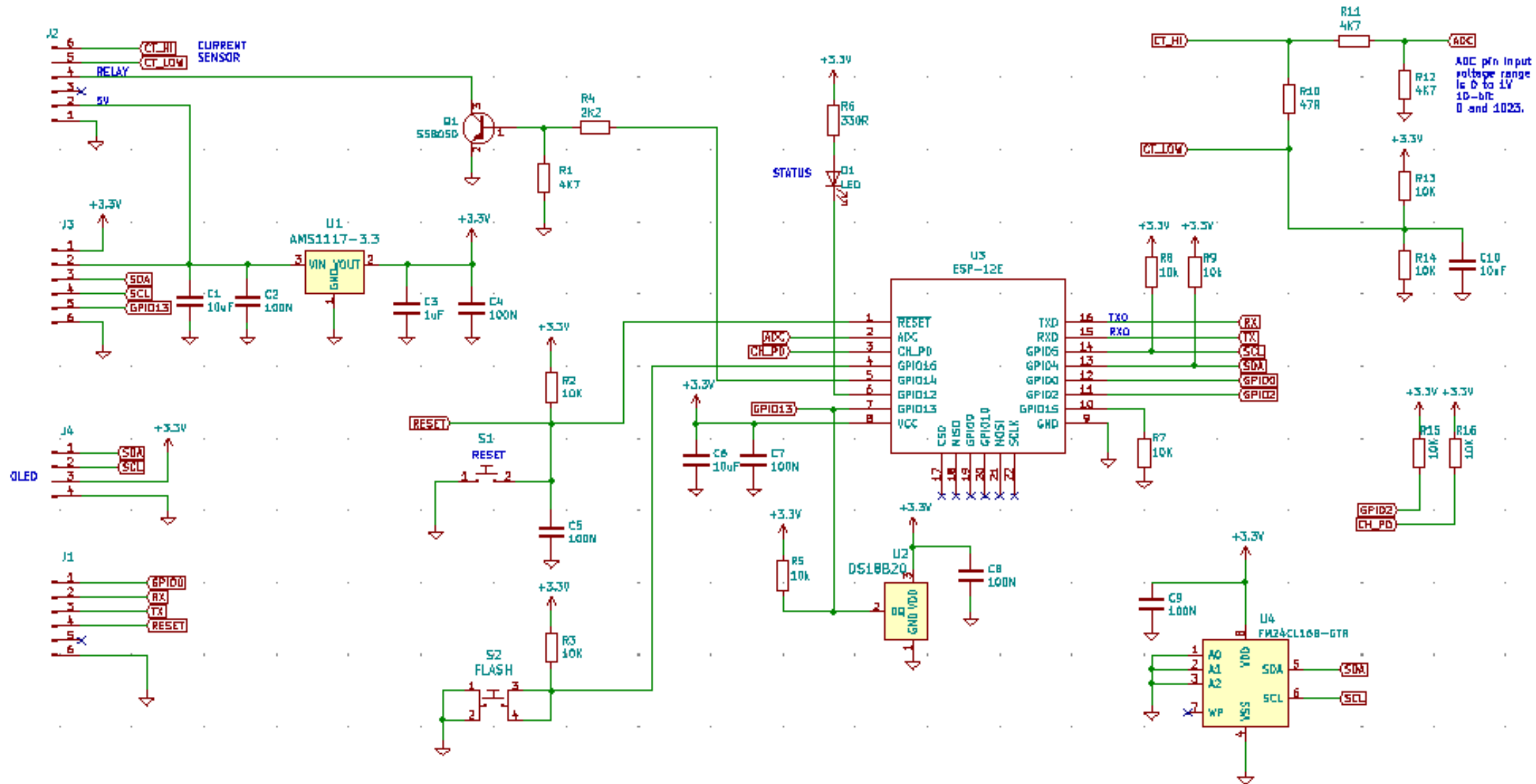
- Geyser Timer Schedule
- Power Consumption Monitoring and Utility Billing on a your own property or a guest house or condo
- Pool Timer
- Under Floor Heating Schedule controller
- Any device that requires an Intelligent Timer

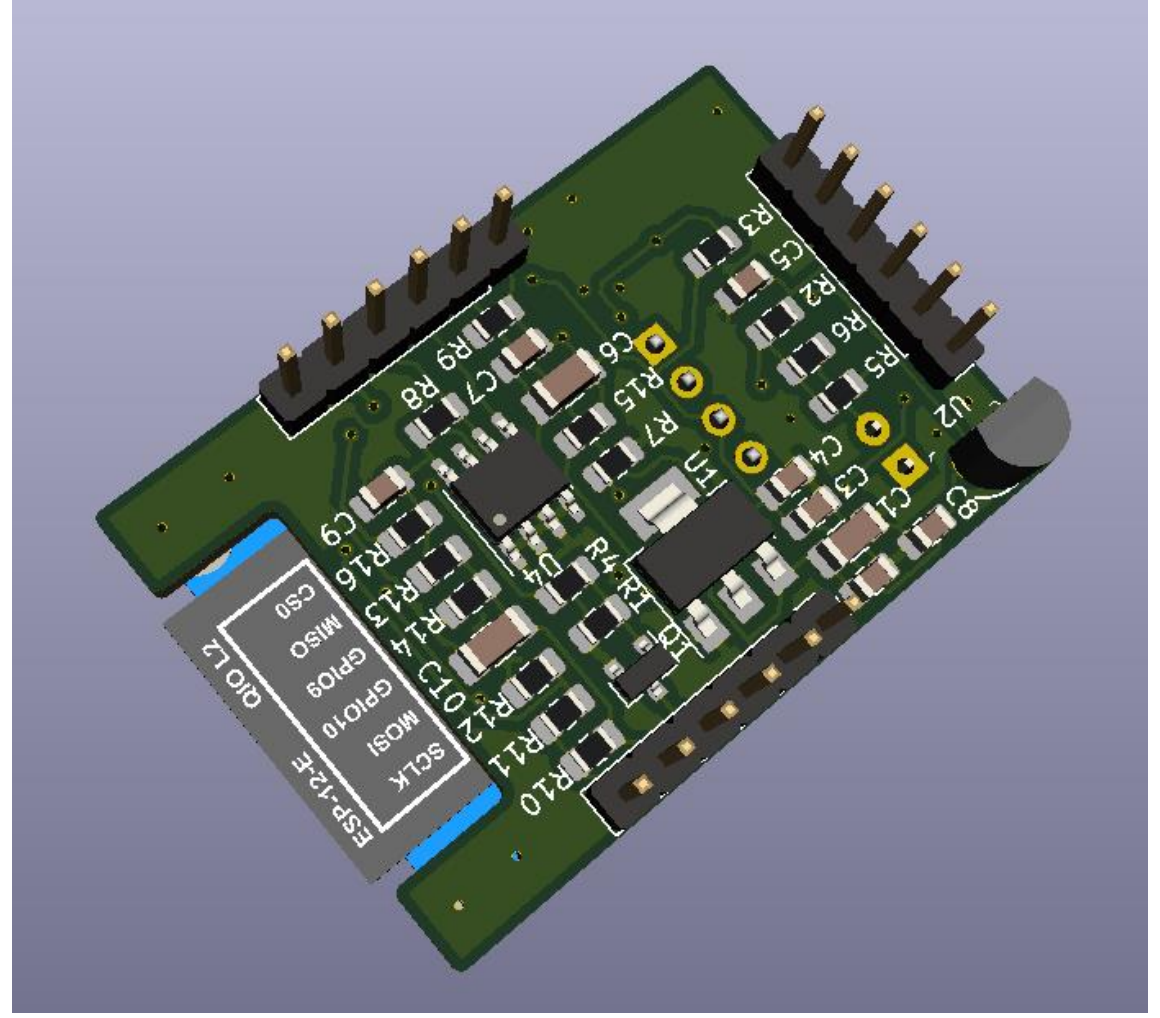
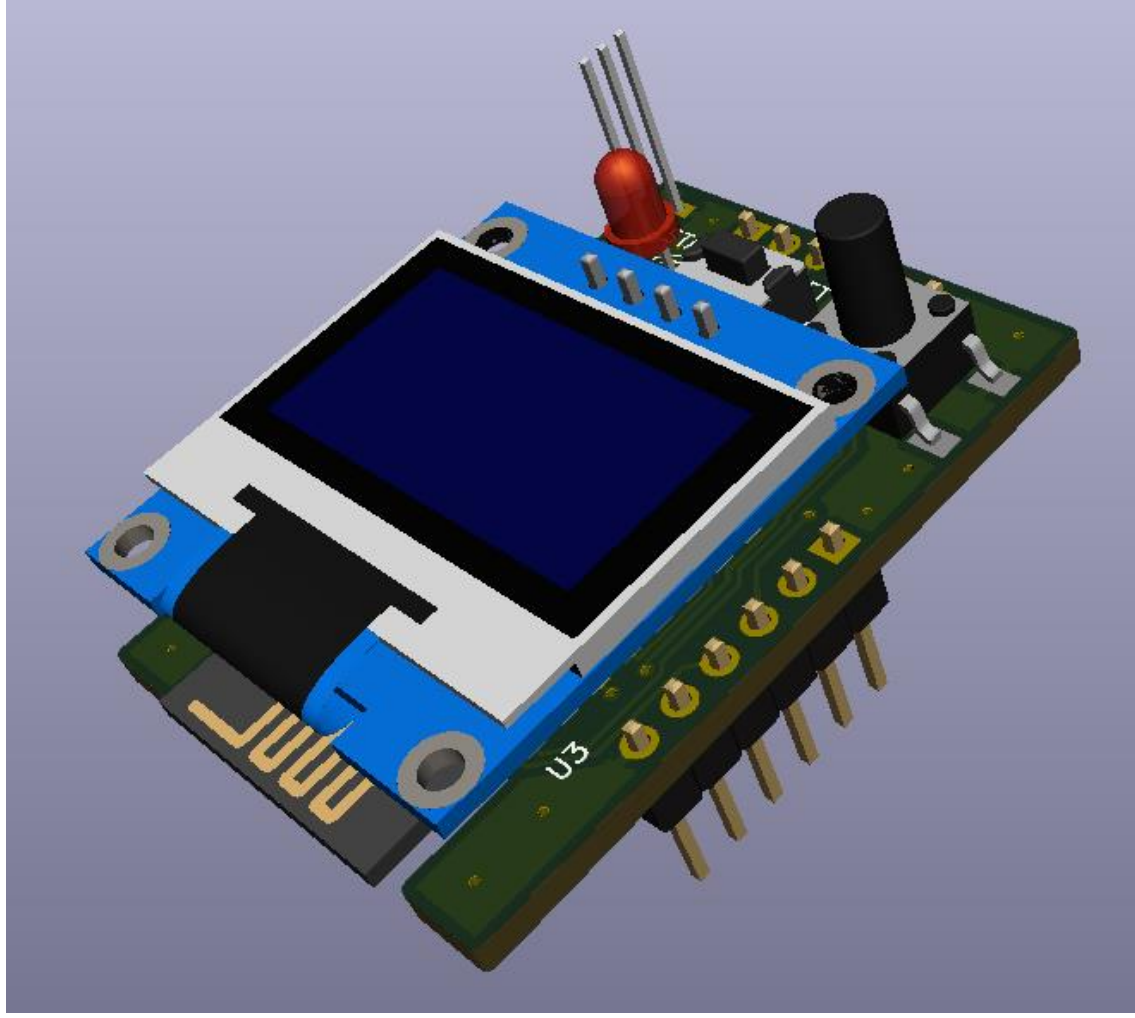
BUILD YOUR OWN

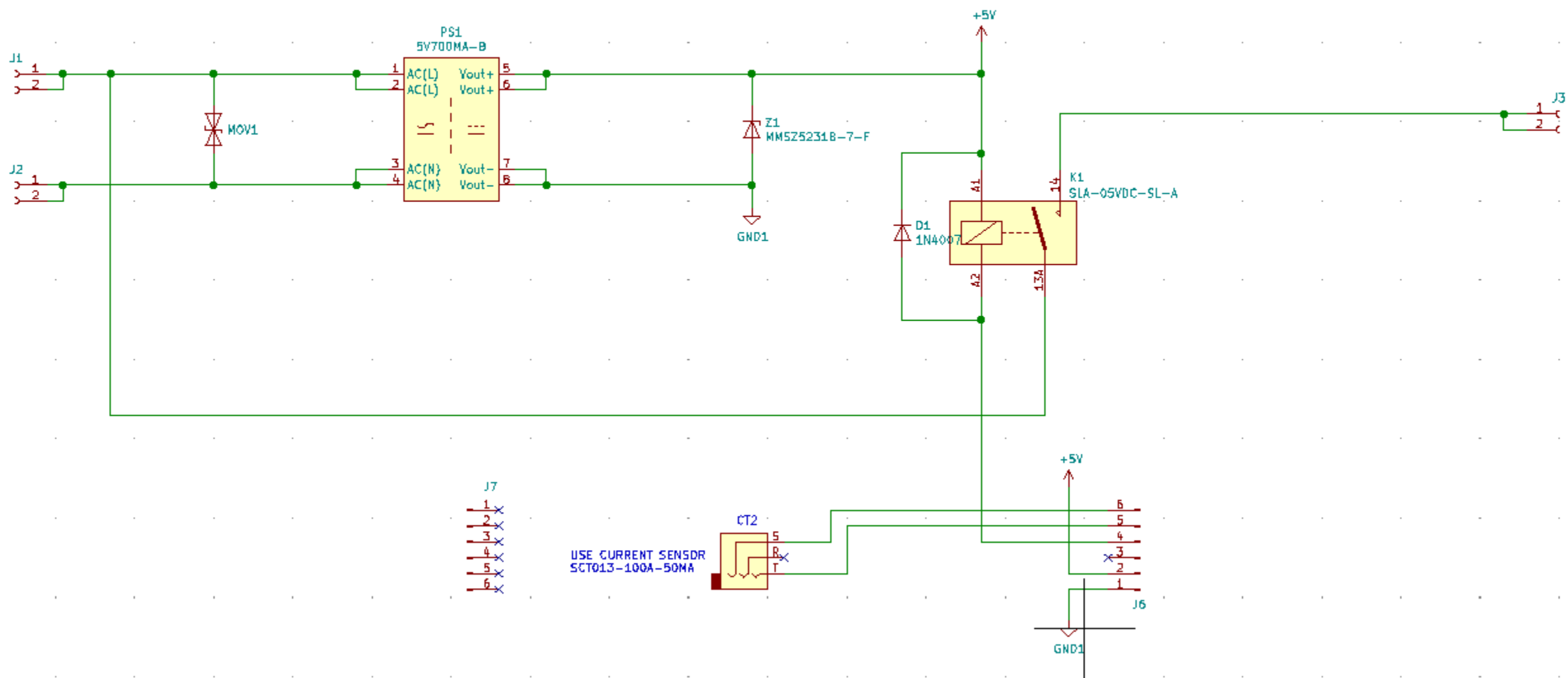
Blank boards or Built up units are available if you are interested in supporting my project. The design is complex, you will need a certain level of skill in Electronics and be familiar with writing code using the Arduino IDE and using Blynk

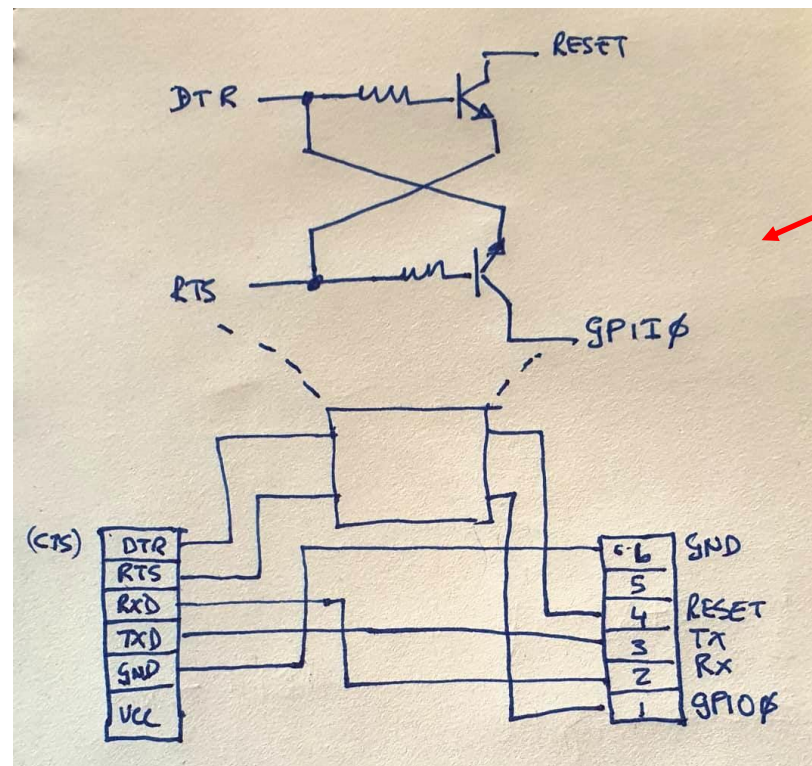
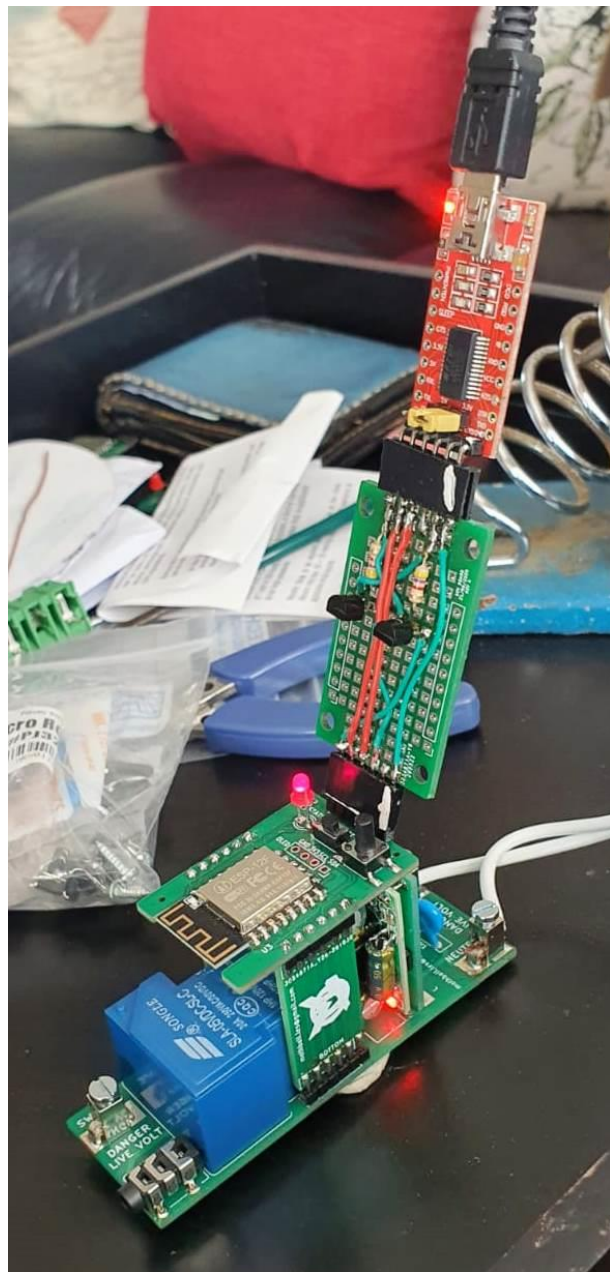
BASIC FUNCTIONS

- WiFi Manager to remotely setup your WiFi Credentials / Blynk Token from your smartphone
- AM & PM schedule programming
- Day of the week programming
- Holiday Mode (Ever forget to switch off your geyser before going on holiday or on a business trip?)
- Need to shower shortly? Override to force the Geyser ON
- Visual feedback on health status and schedule times
- No internal battery required. All schedules are cloud based.
- Immune to a Load Shedding (Power Blackout). In SA we have load shedding that can last anywhere from 1hr to 4hr/day....
- Connected to your home WiFi. (There is a RTC based system in the pipeline for applications where there is no WiFi Connectivity)
- Push Notification. Add the Blynk notification Widget or use the Push Safer API
- On board OLED display. This will only mimic some of the details already displayed on your smartphone and is not essential.
- Smart power consumption monitoring using a non evasive Current Clamp
- Real time power consumption - calculated every second
- Terminal Widget to display the past 12 months usage
- Tariff Slider to set up your KWH/Cost. The data will automatically update every KWH used
- Comparative Power consumption display for Today and Yesterday
- Super Chart – Real time graphic display of your power consumption. The chart is updated every second, the information can also be downloaded as an Excel CVS file for advanced analysis.
- OTA BIN File updates using BitBumper. No need for a physical connection to your computer.



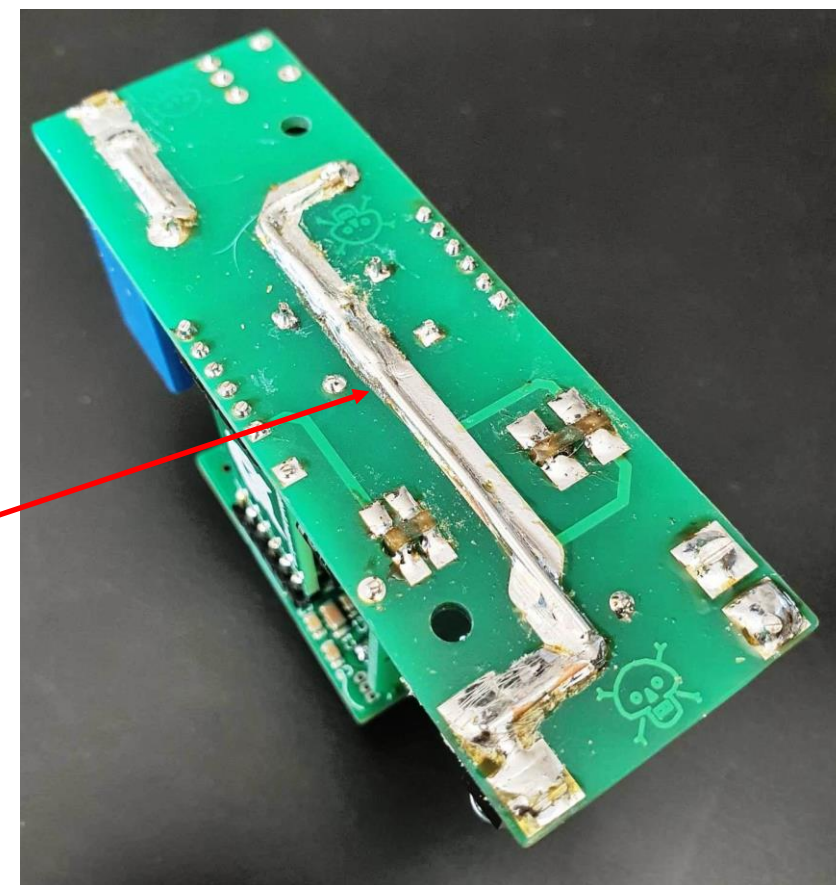






Homemade software download tool.

Tracks beefed up to carry the 30Amp load.

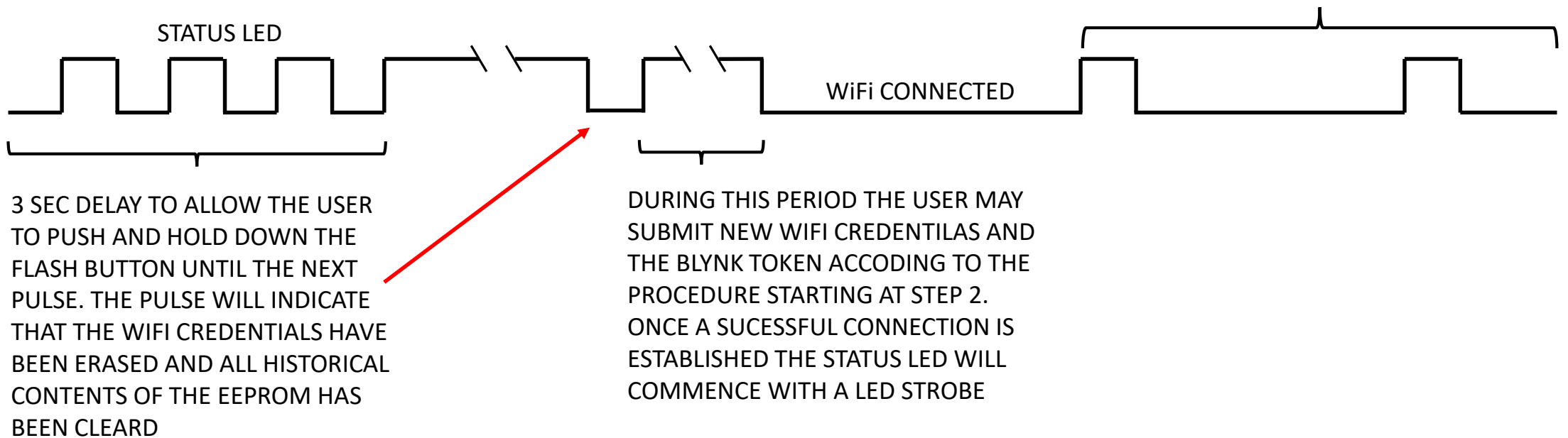


CONFIGURATION

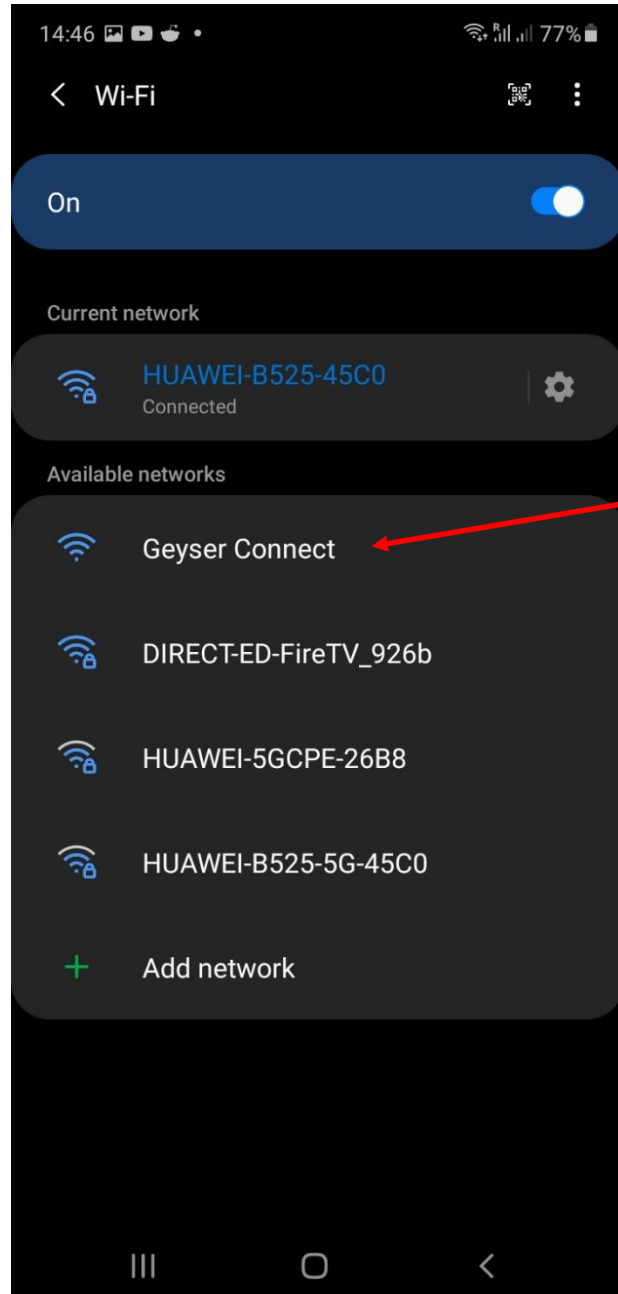
- When your Geyser Timer starts up it will automatically connect to your WiFi if it has previously been set up to do so, however in the event of a new installation go directly to **STEP 2** to program in your Wifi Credentials and Blynk Token using the WiFi Manger
- If you want to changed your Wifi details you will first have to erase your existing particulars using **STEP 1** then continue with the installation steps

STEP 1

TIMING DIAGRAM

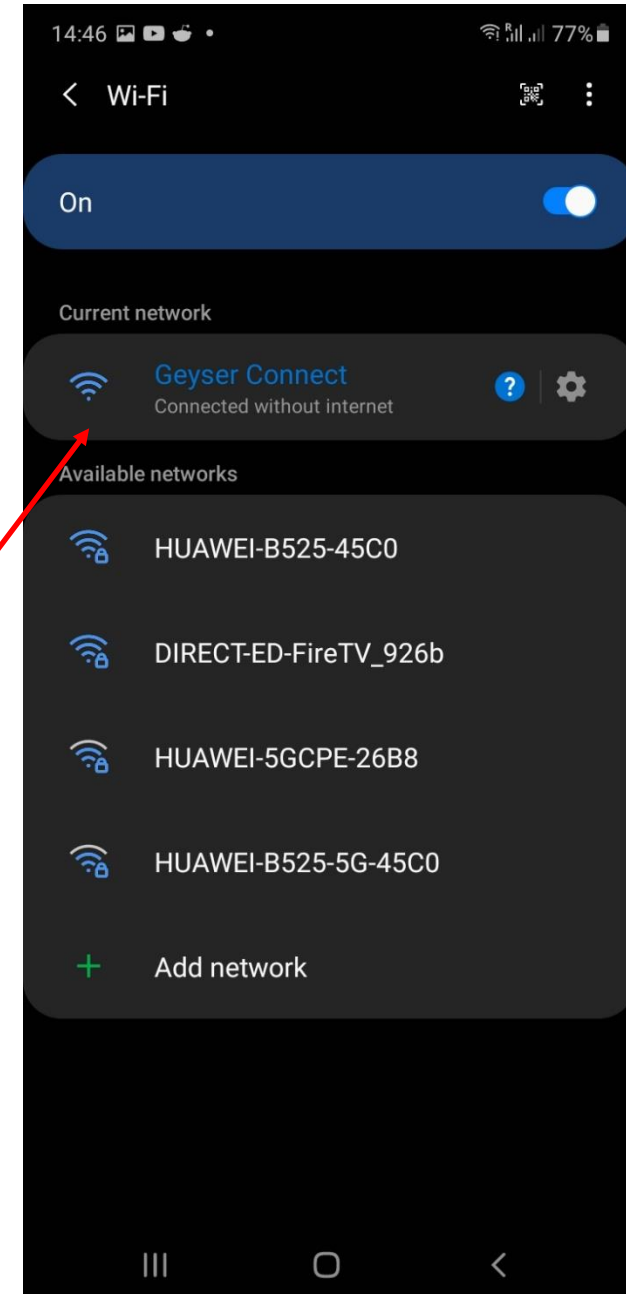


STEP 2

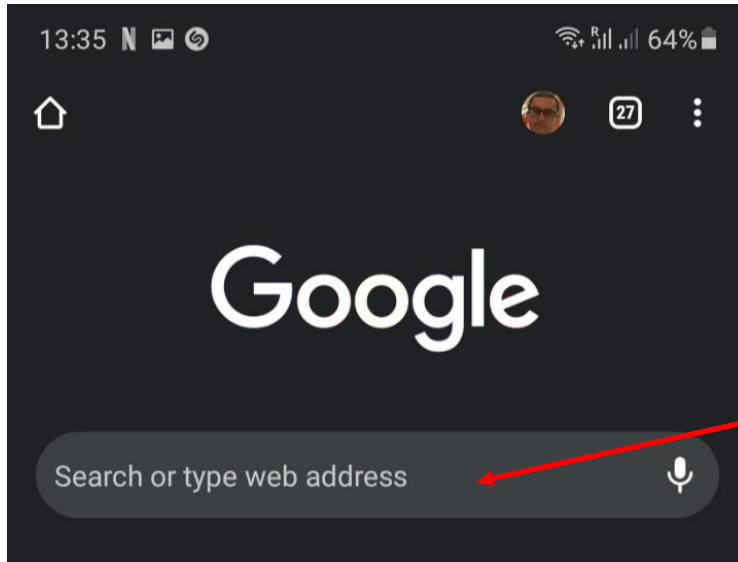


Navigate to the WiFi settings on your phone. Select **"Geyser Connect"**

The Geyser Timer will now act as an Access Point (WiFi hotspot) without an Internet Connection



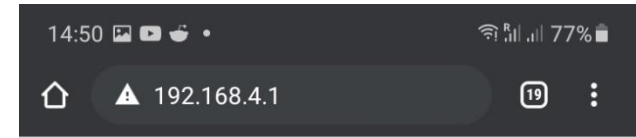
STEP 3

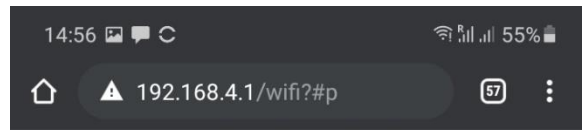


Now using any WiFi enabled device with a browser (computer, phone, tablet) connect to the newly created Access Point as follows:

Open a new tab on your web browser and type in the IP address **192.168.4.1**

A connection to the Geyser Timer will be established and the WiFi Manager configuration portal will pop up in your browser. Select “**Configure WiFi**” to enter your WiFi Credentials and Blynk Token





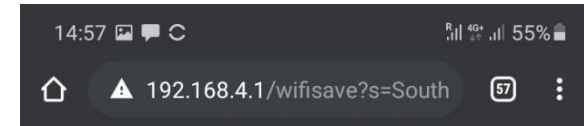
Southpaw	🔒 100%
IoT2	🔒 100%
astra-one	🔒 100%
Migwalla	🔒 84%
DIR-825-0964	🔒 76%
AstraEpic	🔒 60%
NOC	🔒 34%
AstraVenom	🔒 28%
dlink-EC0A	🔒 24%
Benedict	🔒 24%

[Scan](#)

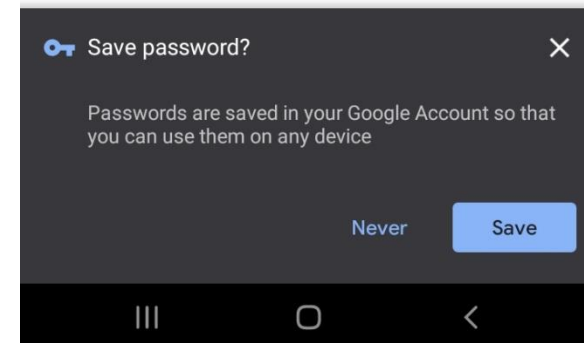


In the configuration portal
select your desired WiFi
Network from the list
presented

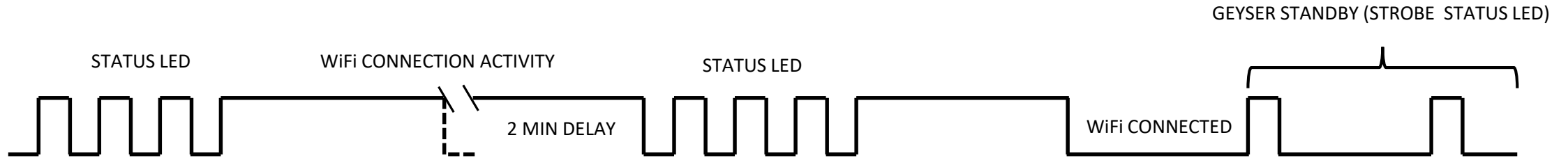
Enter your WiFi Password
and Blynk Token and Save



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



THE GEYSER WILL NOW AUTOMATICALLY TRY AND CONNECT



TIMING DIAGRAM

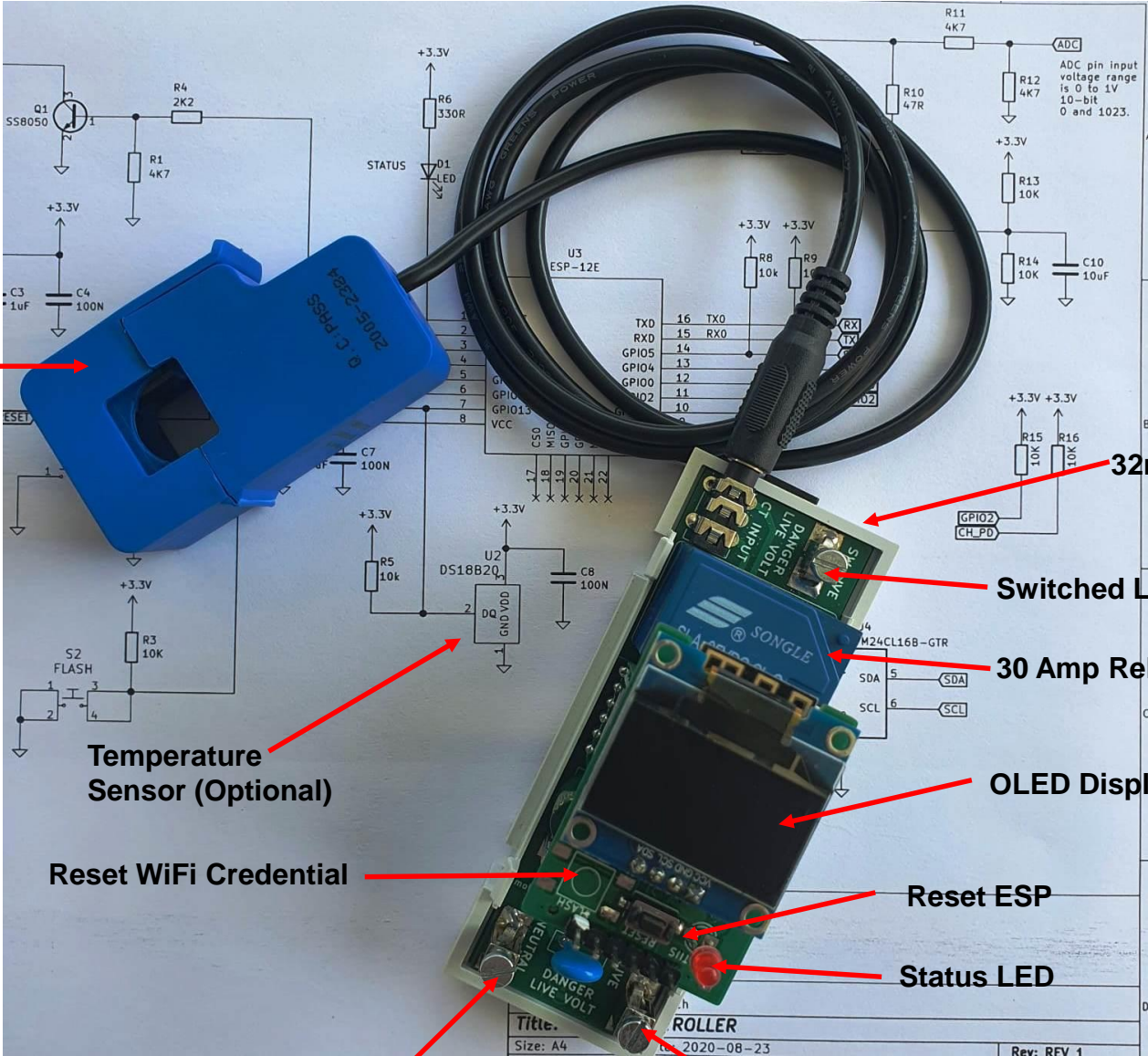
If the Geyser Timer fails to connect on the first round at the point of the dotted line, it will wait for 2 min and then try again. The reasons for a failed connection could be:

- There was a power failure and the Geyser Timer tried to connect to your WiFi before your ROUTER established an Internet Connection
- There could also be a situation where you switched OFF your WiFi. The Geyser Timer will automatically detect that the WiFi connection has been dropped and will try and re-establish a connection, once your ROUTER has been switched back ON

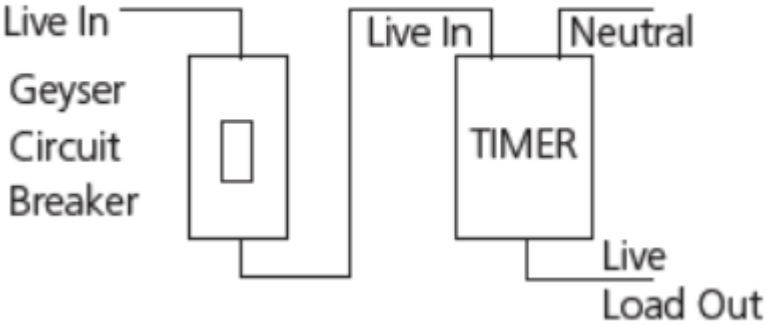
If the it still fails to connect after several minutes then check the following:

- Your WiFi is OFF - Switch ON your ROUTER and wait for it to connect to the Internet.
- You did not enter your WiFi Credentials correctly. – Go to Step 1

HARDWARE



Wiring diagram



Current Clamp Transformer (Clamp to LIVE lead on the Mains Input)

Temperature Sensor (Optional)

Reset WiFi Credential

NEUTRAL from BUS BAR

32mm DIN Rail Housing

Switched LIVE to Geyser

30 Amp Relay

OLED Display

Reset ESP

Status LED

LIVE IN from Geyser CB BAR

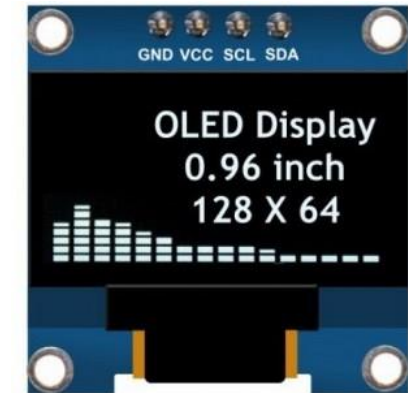
HARDWARE

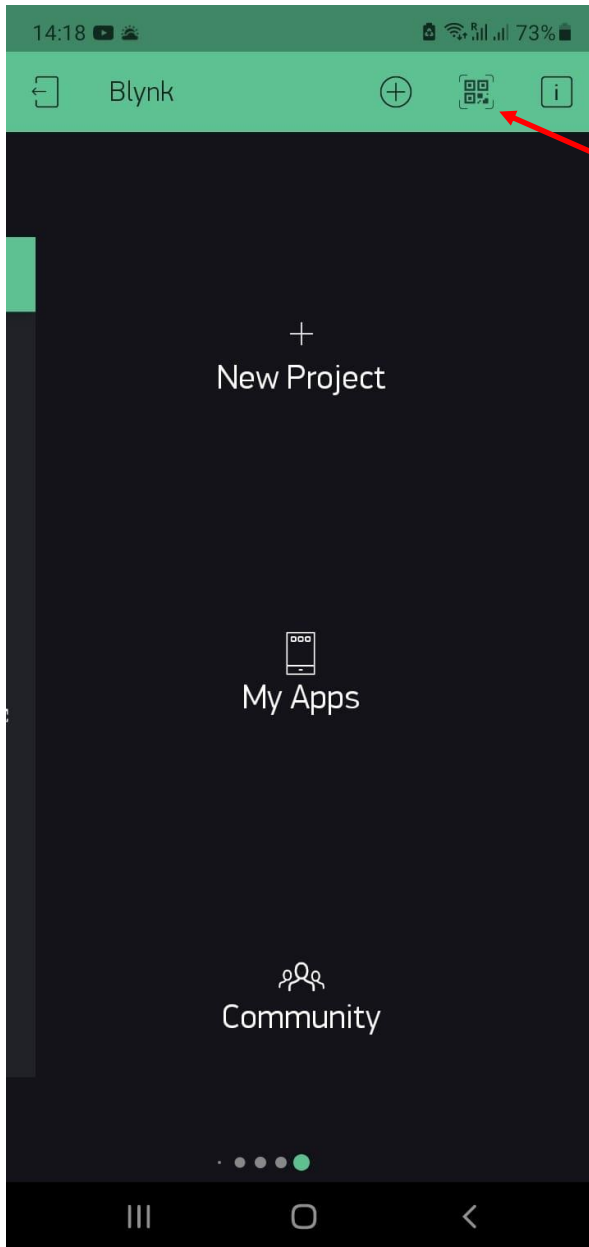


To take advantage of the advanced features of the Geyser Timer. The Temperature sensor (DS18B20) mounted from below and Current Clamp (SCT013-100A-50MA) should be installed. The OLED is optional and a nice to have. To view the TEMP Sensor add a Value Display Widget and set to Virtual Pin 23



A WORD OF CAUTION, I have seen some OLED's that have the **GND** and **VCC** swapped. The **PIN** configuration should map **GND VCC SCL SDA**





CLONING

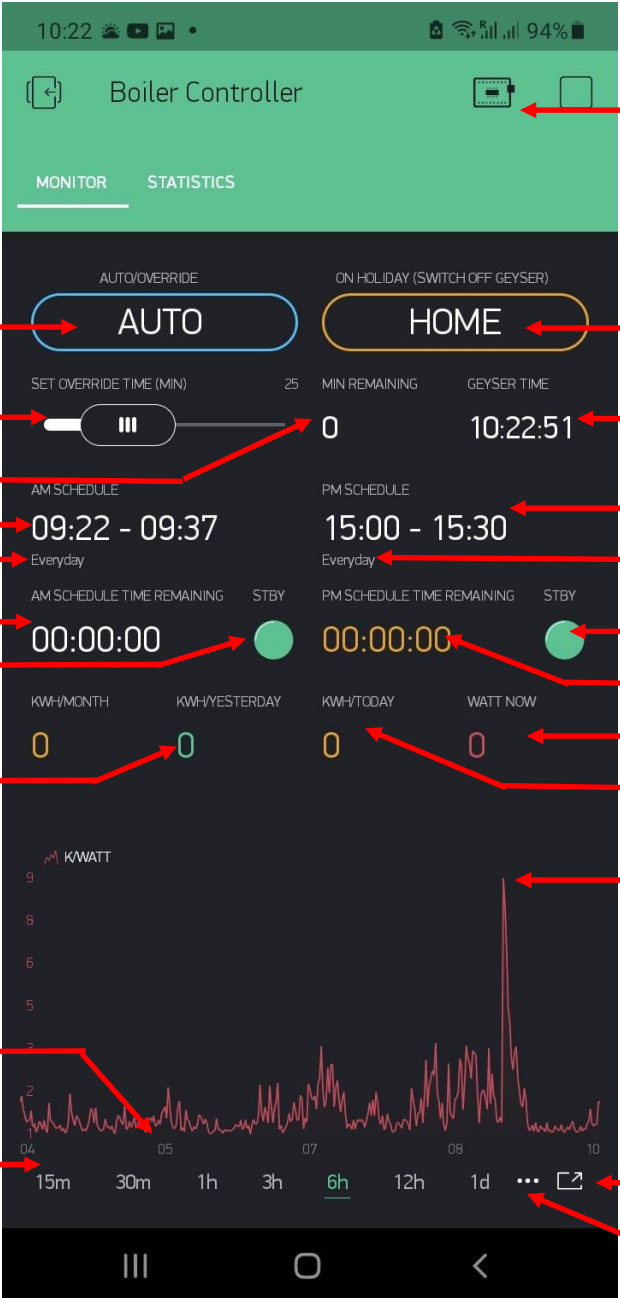
Press QR button in Projects gallery and scan the QR Code. The Geyser Controller project will be cloned, containing all the widgets and settings.
Note the user should make sure they have enough energy points to create the project.

BLYNK QR CODE





Geyser Timer **OFF LINE** indication



Geyser Timer **ON LINE** indicator

Override to force the geyser ON for a predefined time

Holiday Mode (Ever forget to switch off your geyser before going on holiday or on a business trip? Then this is your lifesaver)

Override predefined time setting (10 – 60 min)

Time Display (Acts as a heartbeat/health status)

Override time remaining

AM Schedule

PM Schedule

AM Day Schedule

PM Day Schedule

AM ON Time remaining

PM Status LED (**NOTE 1**)

AM Status LED

PM ON Time remaining

Power Now (Updated every second)

Snapshot of KWH used yesterday

KWH used today (Updated for every 1000 Watt used)

Super Chart – Real time display of your power statistics over time (Updated every second) The display can also be scrolled Left OR Right to view your history

Event Time

Select the resolution to display

Select to display Super Chart in Landscape

Select to Erase Data OR to Download a CVS File

NOTE 1
GREEN = Geyser OFF (STANDBY)
RED = Geyser ON
YELLOW = Geyser Lockdown
PINK = ERROR (Schedule out of SPEC)

The screenshot displays the 'ASTRA' application interface on a mobile device. The top status bar shows the time as 15:04 and battery at 83%. The app header includes a back icon, the name 'ASTRA', and a camera icon. Below the header are two tabs: 'MONITOR' and 'STATS/BILLING', with the latter being the active tab.

The main content area is titled 'TERMINAL' and contains the following information:

- Tariff Cost/KWH = R1.84**: Indicated by a red arrow labeled 'KWH Tariff Cost'.
- Monthly Usage Table**: A table showing monthly KWH usage and cost. A red arrow points to the 'April' row, labeled 'Arrow to indicate the current month'.

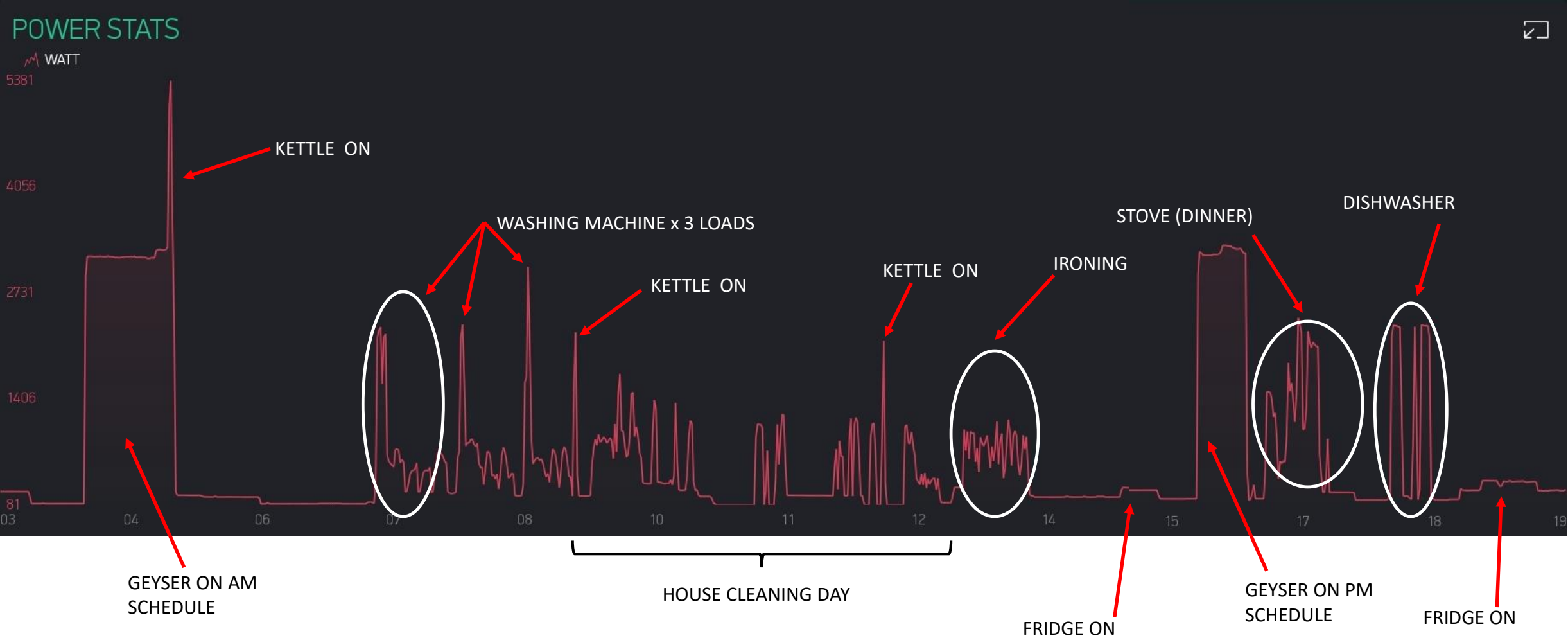
MONTH	KWH/M	COST
January	000	R0.00
February	000	R0.00
March	000	R0.00
April	151	R277.84 <--
May	000	R0.00
June	000	R0.00
July	000	R0.00
August	000	R0.00
September	000	R0.00
October	000	R0.00
November	000	R0.00
December	000	R0.00
- Daily Usage Table**: A table showing daily KWH usage. A red arrow points to the '06' value, labeled 'Current Day'.

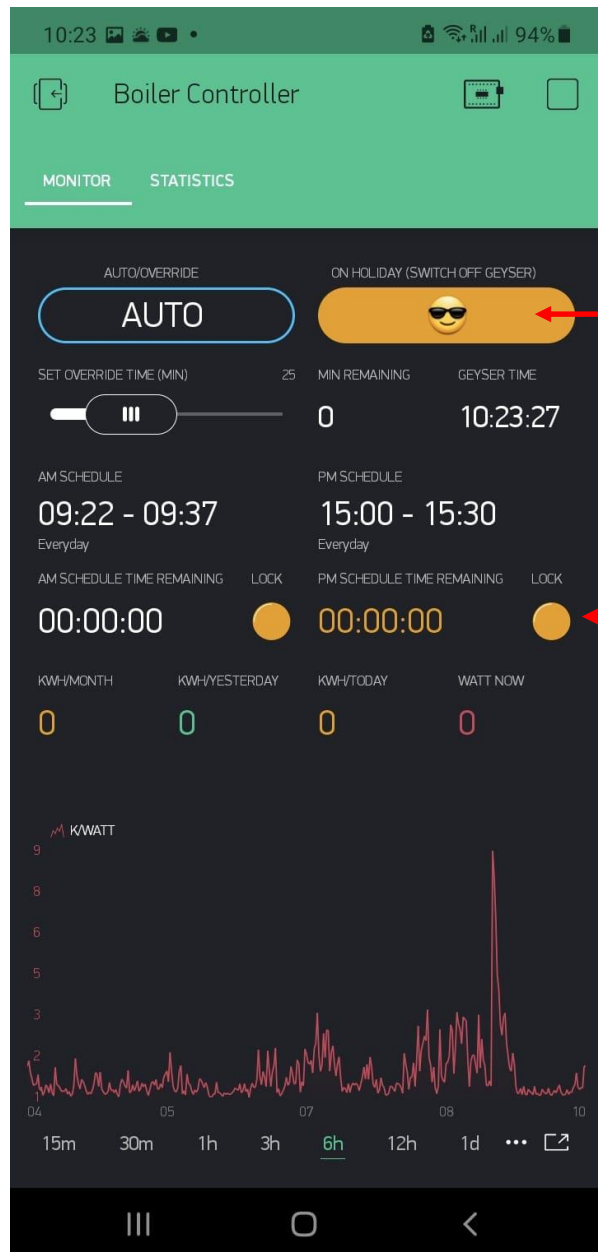
DAILY KWH USAGE						
S	M	T	W	T	F	S
07	06	00	08	08	10	08
- Connection Info**: IP: 192.168.1.167, SOFTWARE V2.0 26/04/2021, DESIGNED BY: mothball.les@gmail.com. A red arrow points to this section, labeled 'IP PORT Connection (see BITBUMPER for details)'.
- Billing Rate Slider**: A slider control for the 'BILLING RATE/KWH' with a value of 184. A red arrow points to it, labeled 'KWH Tariff Cost Setting'.
- Reset Button**: A large red circular button labeled 'RESET'. A red arrow points to it, labeled 'RESET All recorded history'.

Additional annotations on the right side of the image include:

- 'KWH Monthly usage history (Updated every hour)' pointing to the monthly usage table.
- 'KWH Daily usage history (Updated every hour)' pointing to the daily usage table.

Landscape Display



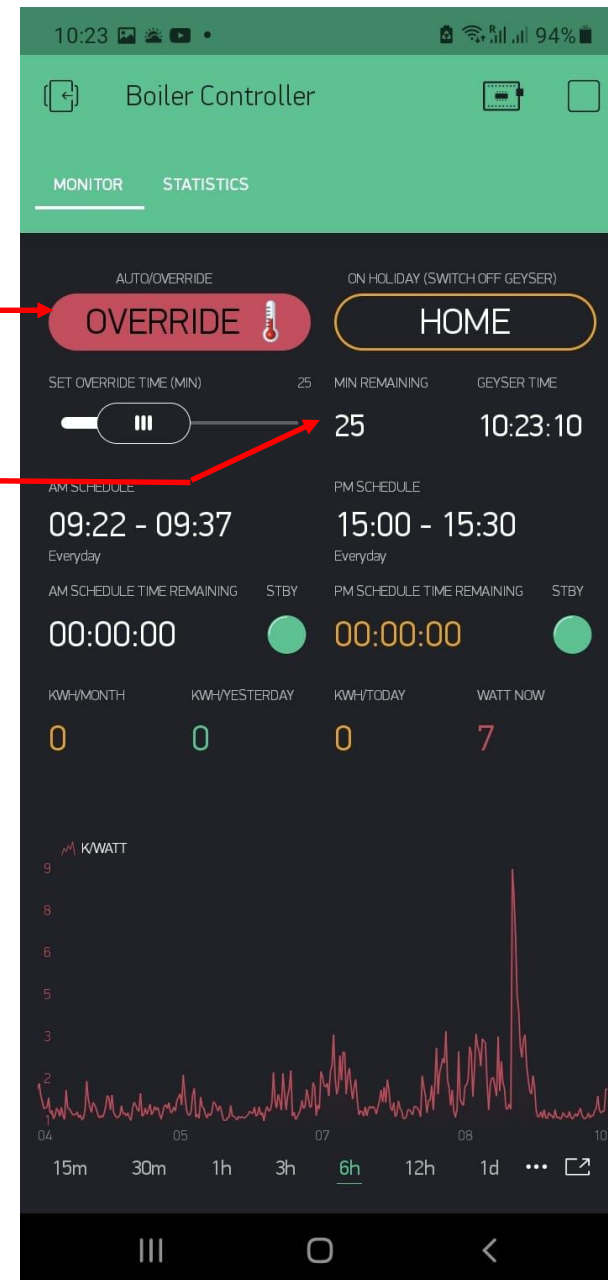


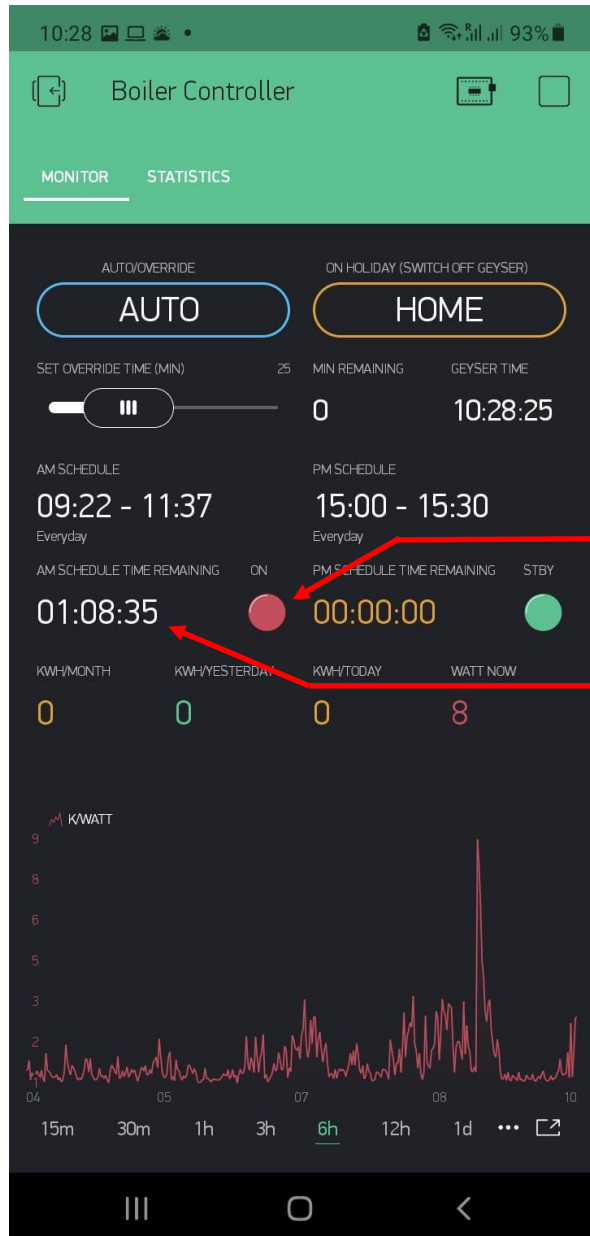
Holiday Mode Selected

Holiday Mode ON Indicator (LOCK)

Override Selected (Geyser ON)

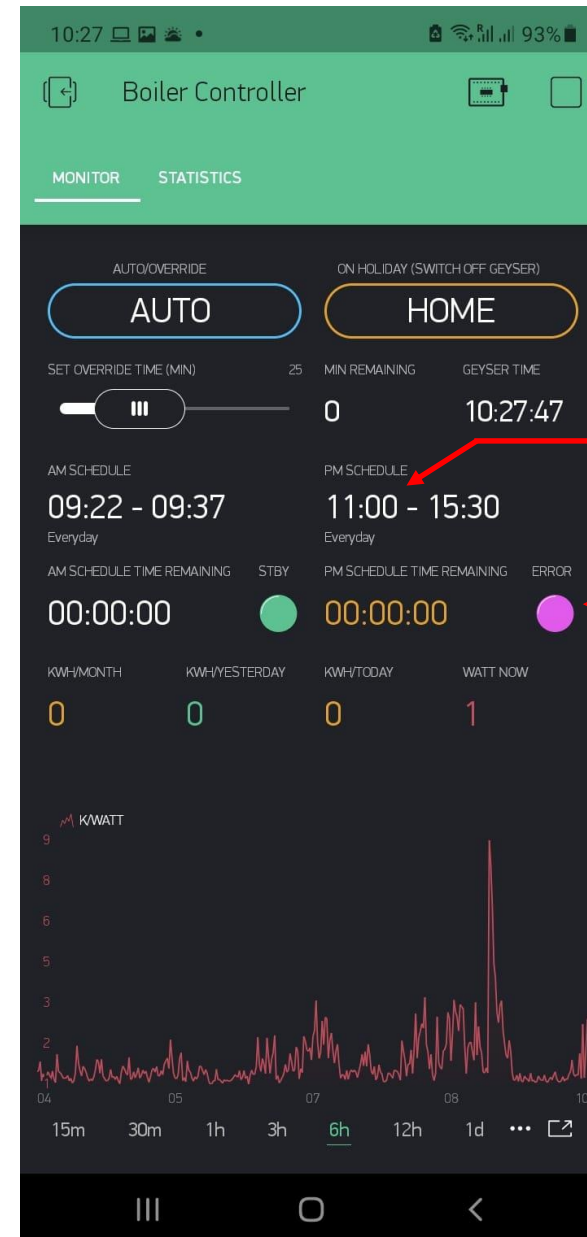
Override MIN remaining





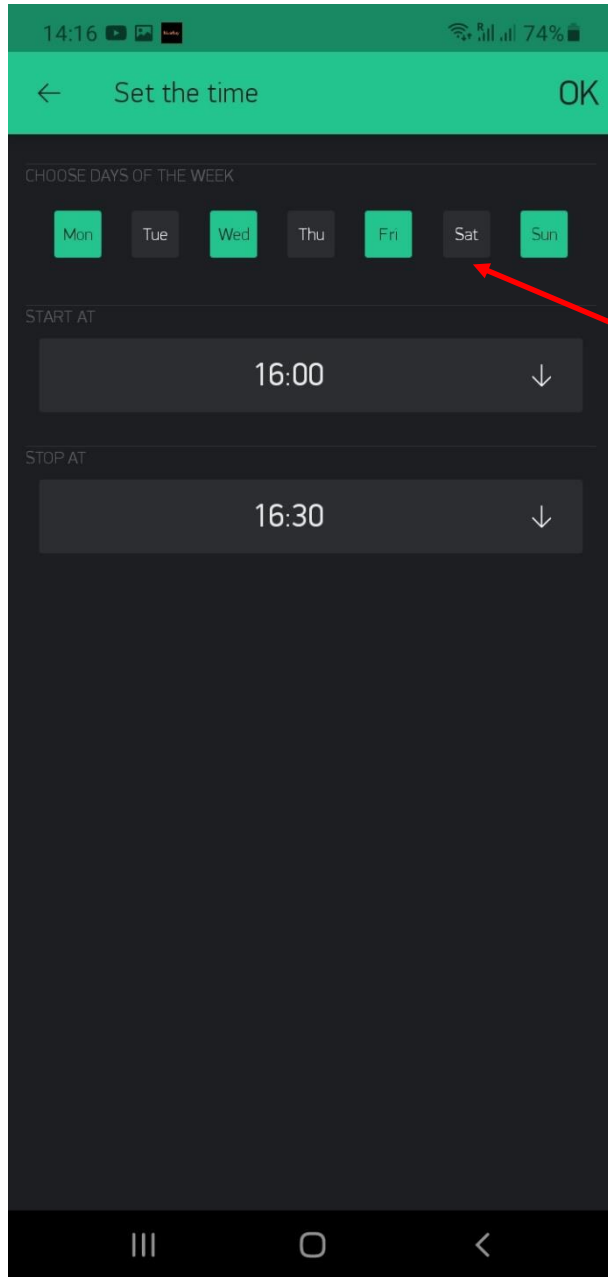
Geyser ON

Geyser ON Time remaining

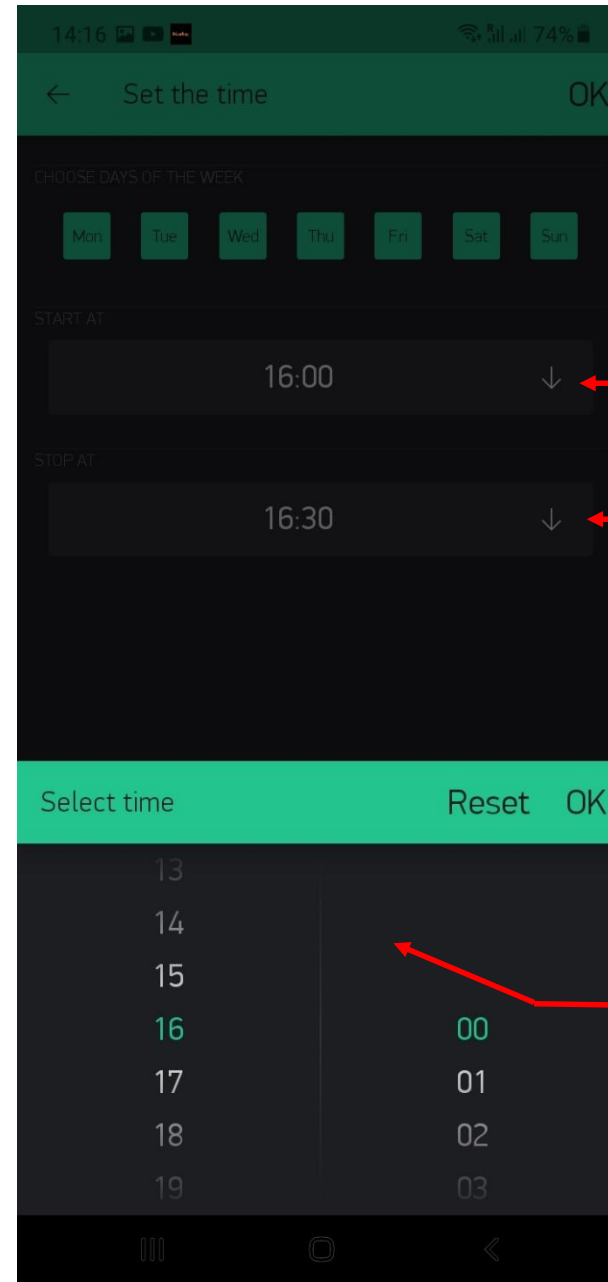


Start Time ERROR (Must be PM Time)

Schedule Time ERROR



Day Schedule Select/Deselect



Start Time Select

Stop Time Select

Time Schedule Select

OTA BIN FILE UPDATING

NOTE: Only when you have your WiFi Manager working will the BitBumper App work

See <https://1st.bitbumper.de/ota-firmware-update-tool-for-esp8266/>

Note: The BIN file must reside in the same file location as the BitBumper EXE

The IP Address of your Geyser Controller connection (See your TERMINAL Display for the IP setting)

Initially just type in 12345 as your Local Port

Flash the new software

BitBumper ESP8266 Flash

Filename: WiFi_Geyser_Controller.ino.nodemcu.bin

ESP8266 Tcplp @: 192.168.1.179 Port: 8266

Local Port: 29289 ☒ random ☐ verbose

Flash

bitbumper software solutions
visit www.bitbumper.de
Heidelberg, Germany

Successful software update

```
12:23:41 246 **** Flash ESP with
[WiFi_Geyser_Controller.ino.nodemcu.bin]
12:23:41 246 Send invitation to ESP @ 192.168.1.179 with MD5
[73f1674acee40644518a7548ed6dae1f]
12:23:41 741 Wait for device data channel
12:23:42 036 Sending data...
12:23:51 421 ****
12:23:51 421 *** successfull updated ***
12:23:51 421 *** device will reboot ***
12:23:51 421 ****
```

CODE

The code is structured over several TAB's categorized for ease of use and reusability.

WiFi_Geyser_Controller - BLYNK_TIMER_AM.ino | Arduino 1.8.13 (Windows Store 1.8.42.0)

File Edit Sketch Tools Help

WiFi_Geyser_Controller BLYNK BLYNK_TIMER_AM BLYNK_TIMER_PM CONNECT DS18B20 EEPROM EMON KWATT_CALC OLED PUSH_SAFER ROLLING_AVERAGE RUN_SCHEDULE TERMINAL UTILITIES WiFi_AAN

```
1 // https://github.com/PaulStoffregen/Time
2
3 //.....
4 //AM_Schedule time decode
5 //.....
6 BLYNK_WRITE(V4) {
7   TimeInputParam t(param);
8   bool time_error_flag = false;
9   long startTimeInSecs;
10  long stopTimeInSecs;
11  long nowseconds;
12  byte bytes[3];
13  byte day_data[8];
14  byte x;
15
16      startTimeInSecs = param[0].asLong();
17      nowseconds = ((hour() * 3600) + (minute() * 60) + second()); //time right now
18      stopTimeInSecs = param[1].asLong();
19
20      if (startTimeInSecs >= 43200){
21          switch_off_system(1);
22          led_status (1,2); //timer 1 = ERROR
23          time_error_flag = true;
24      }
25      else if (stopTimeInSecs >= 43200){
26          switch_off_system(1);
27          led_status (1,2); //timer 1 = ERROR
28          time_error_flag = true;
29      }
30      else if (startTimeInSecs > stopTimeInSecs){
```

Done Saving.

RODATA : 43904 - / 81920 = constants (global, static) in RAM/HEAP
BSS : 31608 - zeroed variables (global, static) in RAM/HEAP
Sketch uses 422188 bytes (40%) of program storage space. Maximum is 1044464 bytes.
Global variables use 37908 bytes (46%) of dynamic memory, leaving 44012 bytes for local variables. Maximum is 81920 bytes.

NodeMCU 1.0 (ESP-12E Module), 80 MHz, Flash, Legacy (new can return nullptr), All SSL ciphers (most compatible), 4MB (FS:2MB OTA:~1019KB), 2, v2 Lower Memory, Disabled, None, Only Sketch, 115200 on COM8

Type here to search

4:45 PM 2021/04/26

BLYNK VIRTUAL PIN ASSIGNMENT

Note that not all the Blynk Widgets have been assigned for display, some were used for debugging and background information.

V0	WiFi RSSI Value Display Widget
V1	Displays Local Zone Time (Used as Heartbeat)
V2	Override Button: Force the Geyser ON for the pre selected time if no holiday mode selected.
V3	PM Time Input Setting Widget
V4	AM Time Input Setting Widget
V5	LED Geyser ON
V7	Override Time Remaining Value Display Widget
V8	LED Holiday ON
V9	Time remaining when the AM Time Schedule is active
V10	Override Time Slider Setting Widget
V11	Time remaining when the PM Time Schedule is active
V12	LED AM STATUS
V13	Displays accumulated KWH for the current month (Roll over every month)
V14	RMS Current Calculation
V15	Accumulating Raw KWH data updated every second (Engineering Info) The data will be processed every hour
V16	Second counter information. Counts from 0 – 3599 sec (Engineering Info)
V17	KWH Remainder value. (Engineering Info) The remainder from the previous hour is add it to the current KWH Data every hour
V19	Instantaneous Power Reading
V20	Reset all the Historical data (Erase the EEPROM)
V22	LED PM STATUS
V23	Temperature Value Display Widget
V25	Holiday Mode Select (Lock out the schedule)
V26	Displays the last 24hr KWH Reading (Roll over at midnight)
V27	Displays the current accumulated KWH Reading for the day (Updated every hour but only if 1000 Watt/Hr achived)
V31	Power Reading - Assigned to the Super Chart Widget
V40	Cost/KWH Slider Setting Widget
V43	Terminal Widget

KWH CALCULATIONS

The ESP12 Analog pin is read every second applying

The reading is then multiplied with 230V to give the Power reading

The *pwr* readings are then accumulated over hour (saved as *watt_sec*)

After accumulating the data for a period of 1 hr (3600 sec) we then update the KWH

```
Irms = emon1.calcIrms(1480); // EMON Library
```

```
pwr = Irms*230.0;
```

```
watt_sec = watt_sec + (pwr / 3600); //convert Watt/Sec to Watt/Hr
```

```
void Update_Kwatt_HR (float watt_hr_value)
```

WORKING EXAMPLE

REMAINDER = 978 (From the previous Hr)

RAW KWH/DATA = 2927 (Accumulated RAW KWH Data over the last hour)

COUNT = 3598 (ROLL OVER at 3599 = 1hr)

KWH/TODAY = 1

KWH/MONTH = 172

ROLL OVER AT COUNT 3599

There is a small discrepancy in the calculation as I did not quite capture the screen shot at the roll over

$$2927 + 978 = 3905$$

Enter the *for (int i = 0; i <= 15; i++)* loop to calculate the KWH's

3905 - 1000 = 2905 (Add 1 to KWH Count)

2905 - 1000 = 1905 (Add 1 to KWH Count)

1905 - 1000 = 905 (Add 1 to KWH Count)

905 - 1000 = Result less than 1000 save 905 and exit the *for loop*

Add 3 KWH to the account

KWH/MONTH	KWH/YESTERDAY	KWH/TODAY	WATT NOW
172	8	1	3052
COUNT	RAW KWH DATA	REMAINDER	AMP
3595	2927.15	978	13.27

ROLL OVER

KWH/MONTH	KWH/YESTERDAY	KWH/TODAY	WATT NOW
175	8	4	3042
COUNT	RAW KWH DATA	REMAINDER	AMP
3	2.54	908	13.23

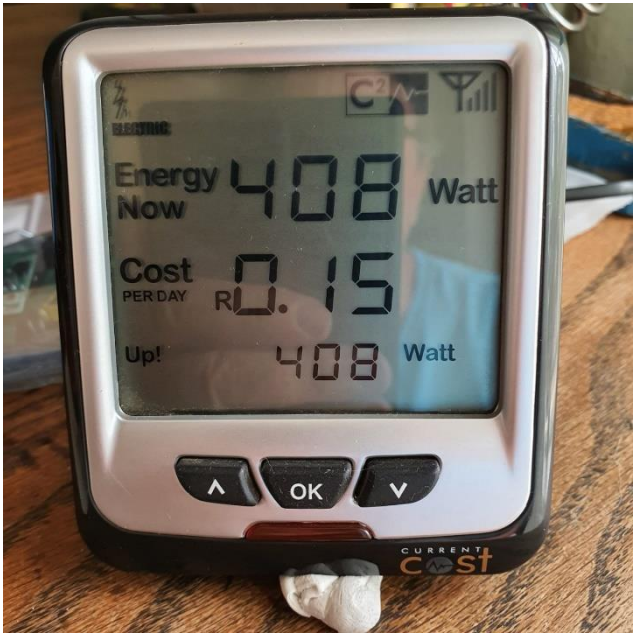
The KWH/TODAY will increase from 1 to 4

3 K/Watt added to the Month Total

The COUNT was reset to start again

CALIBRATION

I used a <http://www.currentcost.com/> meter and a 3 Bar Element heater rated at 1500W (500W/Bar) to determine the calibrate factor for my unit, then by checking my readings against the Current Cost meter I managed to get fairly consistent results. *Note each bar of the heater had its own switch*

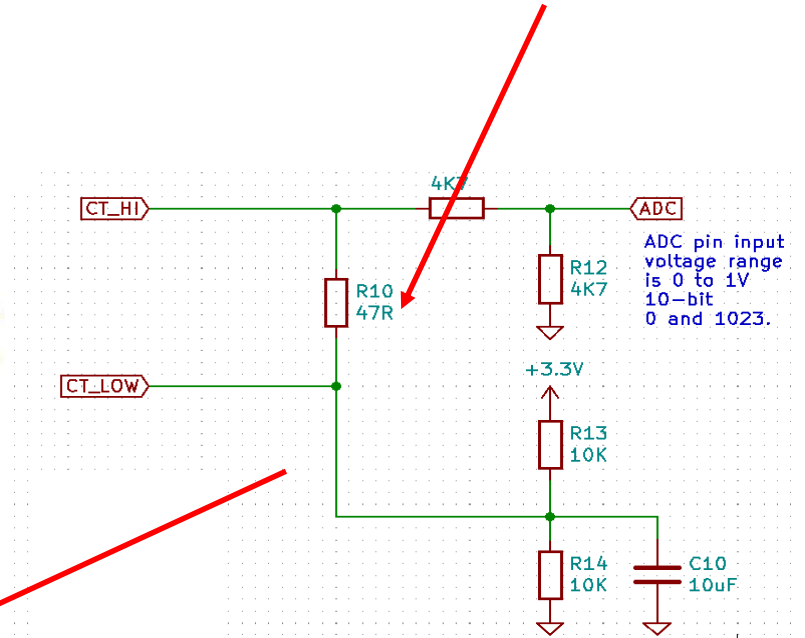
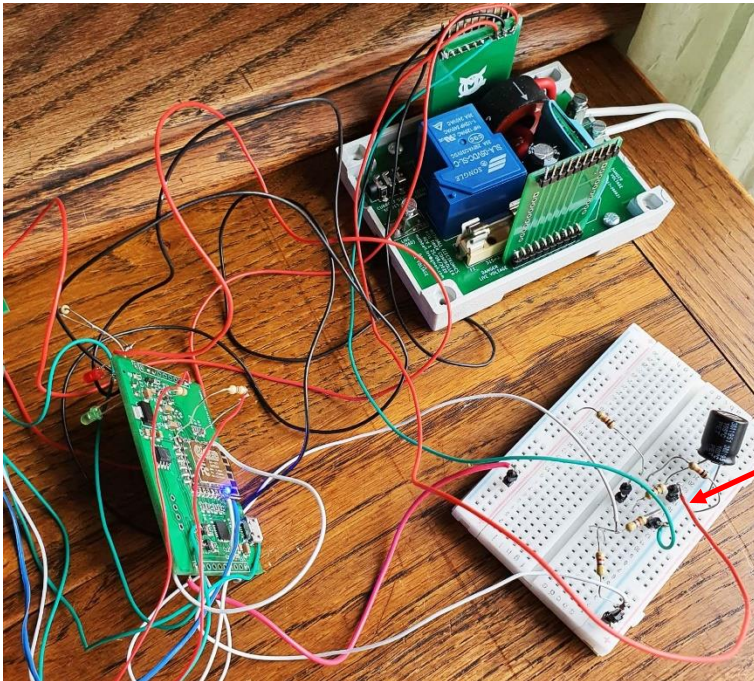


I also compared my Power Consumption readings against my Power Utility Bill (my reading was 283) about 4% discrepancy over a month. One could tweak the calibration factor and then do long term readings to improve the results

Type	Tariff	Meter Number	Read Type	Prev Reading	Pres Reading	Usage
Electricity	100.00	Multilevel	14201442036	31134	31427	293.00

CALIBRATION FACTOR

`emon1.current(A0, 26.37);` // input pin, calibration factor 26.37 for a 1:1800 (SCT013 Current Clamp) CT 100A:50mA and 47ohm burden resistor



Optimizing the network during development