

**HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY OFFICE FOR
INTERNATIONAL STUDY PROGRAM**

SEMESTER: 251



ASSIGNMENT REPORT

INSTRUCTOR:
MR. PHAM CONG THAI

Student Name	ID
Trần Lê Tuấn Anh	2452089
Ngô Đức Anh	2452054
Nguyễn Thành Trường An	2452020

Class ID: CC04

Workload and Team Contribution

No.	Name	Student ID	Workload	Contribution
1	Trần Lê Tuấn Anh	2452089	Report, 4.1, 4.2, 6.1.1, 6.2.3	100%
2	Nguyễn Thành Trường An	2452020	4.5, 4.6, 6.1.1, 6.1.2, 6.1.3, 6.2.2	100%
3	Ngô Đức Anh (nhóm trưởng)	2452054	4.3, 4.4, 6.1.4, 6.1.5, 6.1.6, 6.2.1	100%

1 Objectives

In this project, you are required to use the Altium Designer to design and layout a circuit that synthesize many knowledge that you have learnt in this course. There is something new, but no worries, we support you.

The project will include

- how to design a digital input switches,
- how to use a diode to prevent a short-circuit,
- how to use a transistor generating a high current signal,
- how to use an opamp as a buffer and a low-pass filter before reading an ADC signal, and
- how to connect all the input and output signals to a micro-controller.

2 Specifications

In this project, you aim to design a circuit that is able

- to measure the current of an 220V AC signal,
- to set an address to distinguish with other similar circuits, up to 16,
- to measure the maximum current either up to 5A or up to 10A,
- to send data to a gateway via RS485 or Wifi or Bluetooth,
- (optional) to display on 7 segment LEDs using IC 74HC595.

3 Solution

To fulfill the requirements above, one of the solutions that we can think of is that

- We will use a current sensor [1] to measure the current of the AC signal. The sensor should support to measure up to 5A or up to 10A. There are many current sensors that are available in the market, we recommend you to use TA12 [2] for 5A maximum and TA17 for 10A maxumun. They are cheap and easy to use.
- We will use 4 slide switches to set a board address.
- We will use an IC that can convert from UART signals to RS485 signals [3] for transferring data via RS485.
- We will use a micro-controller (MCU) board, namely ESP32-WROOM-32 [4] as a main processor. ESP32-WROOM-32 is a powerful, generic WiFi and Bluetooth MCU module which is suitable to many IoT application.

Now we list all the part that is required for our circuit. Based on the solution above, the circuit

should include

- A power supply input with the range of 5V - 36V,
- A regulator 3.3V to supply power to the module ESP32-WROOM-32,
- A microcontroller board ESP32-WROOM-32,
- A TA12/TA17 sensor that supports up to 5A or up to 10A,
- Slide switches for a board address,
- LEDs for display status,
- a RS485 circuit,
- and some capacitors for filtering noise.

4 Guidance

In this section, we give you some guidance to draw a circuit with the solution above.

4.1 Design a 3.3V regulator

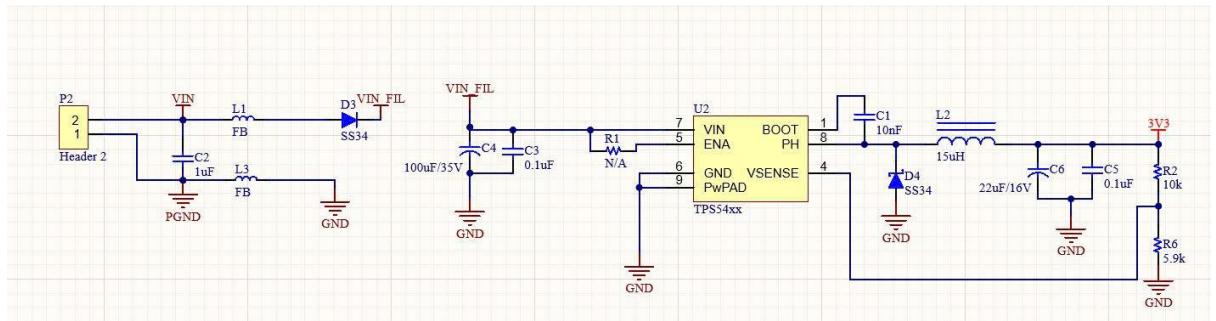


Figure 1.1: Our power supply 3.3V regulator

Figure 1.1 shows a power supply part for this circuit. It generates a 3.3V output for our circuit from input from 5V-36V to 3.3V. On the left, P2 is a input header which 5-36V input is coming. The input power supply goes through a LC circuits (L1, L3 and C2) to filtering the high frequency noises. Then it goes though a diode D3 which is used to prevent the error from input at P2. Finally, we use IC TPS5430 [5] to generate a voltage 3.3V output.

4.2 Design a ESP32-WROOM-32 part

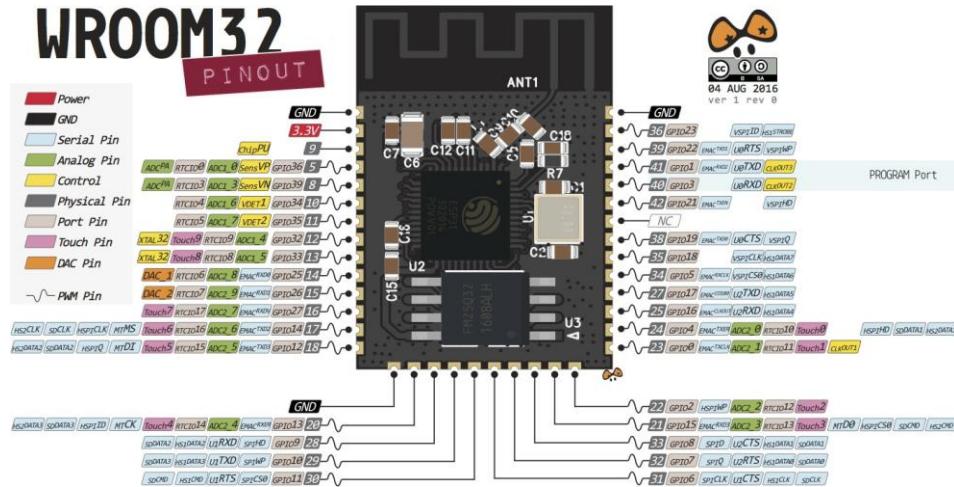


Figure 1.2: ESP32-WROOM-32 pinout

Figure 1.2 shows the pinout of the board ESP32-WROOM-32. It has on board 18 Analog to digital conversions (ADCs). Each ADC is 12 bit SAR technology based. 2 digital to analog conversion (DACs). It integrates 9 touch sensors. For communication, it has 2 UART communications channels, 2 I2C communications interfaces, two I2S channels and one CAN communication interface. It has 16 pulse width modulation channels. It also has a cryptographic hardware acceleration module for various cryptographic algorithms like RSA, AES.

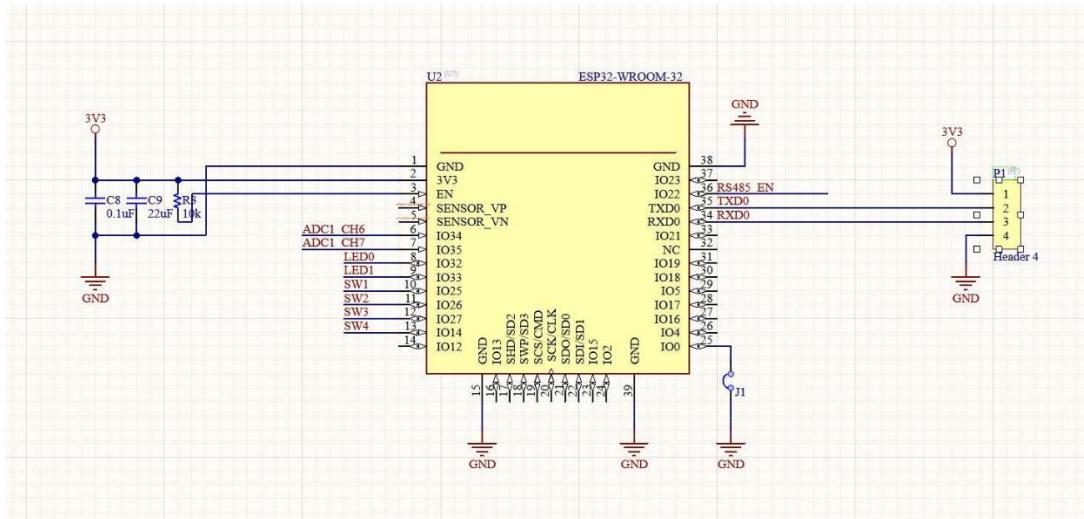


Figure 1.3: Our ESP32-WROOM-32

To fulfill the solution above, we will use 2 pins for ADC inputs, 2 pins for LEDs, 4 pins for switches and 3 pins for RS485 as shown in Figure 1.3.

4.3 Interfacing Slide Switch with an MCU

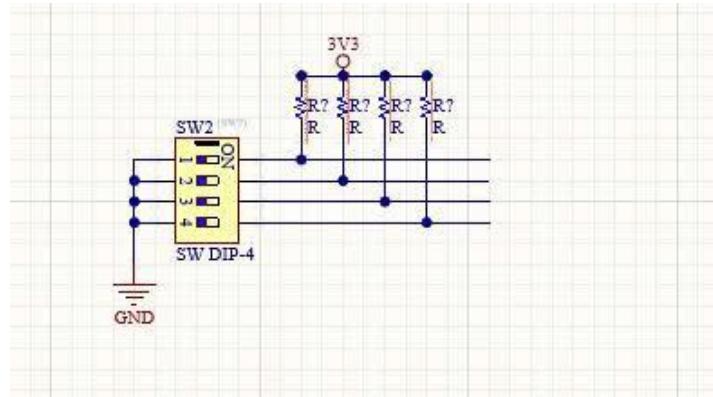


Figure 1.4: Our switches

We use a SW DIP-4 for 4 switches which each switch has two terminals. One terminal connects with ground and the other connects with a resistor to 3.3V power supply and connects to the MCU pins. For resistors you can use 4 single resistors or you can use a RAID including 4 resistors inside.

4.4 Current sensor circuit

In this sensor part, we use two opamps which are packed in one IC LM358. IC LM358 includes two opamps. We use one to create a reference voltage, while we connect the second opamp with two current sensor TA12 and TA17 as shown in Figure 1.5.

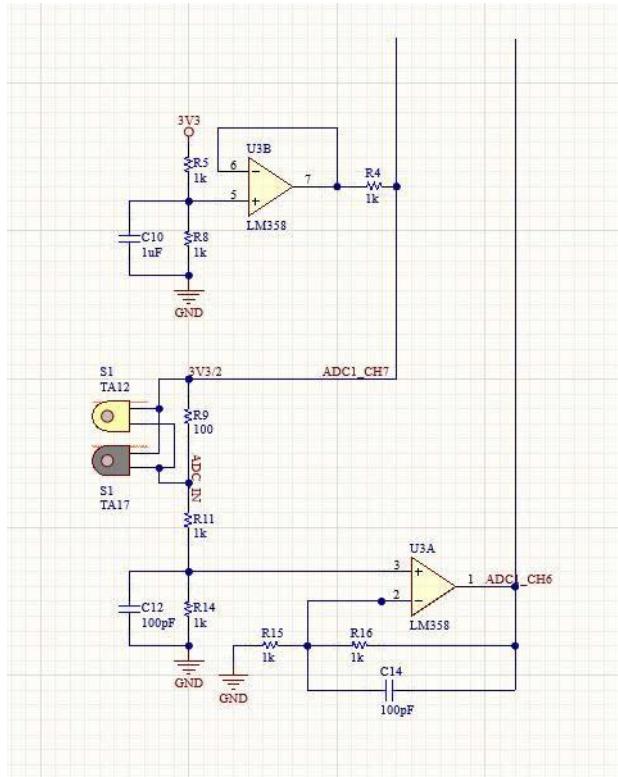


Figure 1.5: Our ADC Input

4.5 Design a RS-485 part

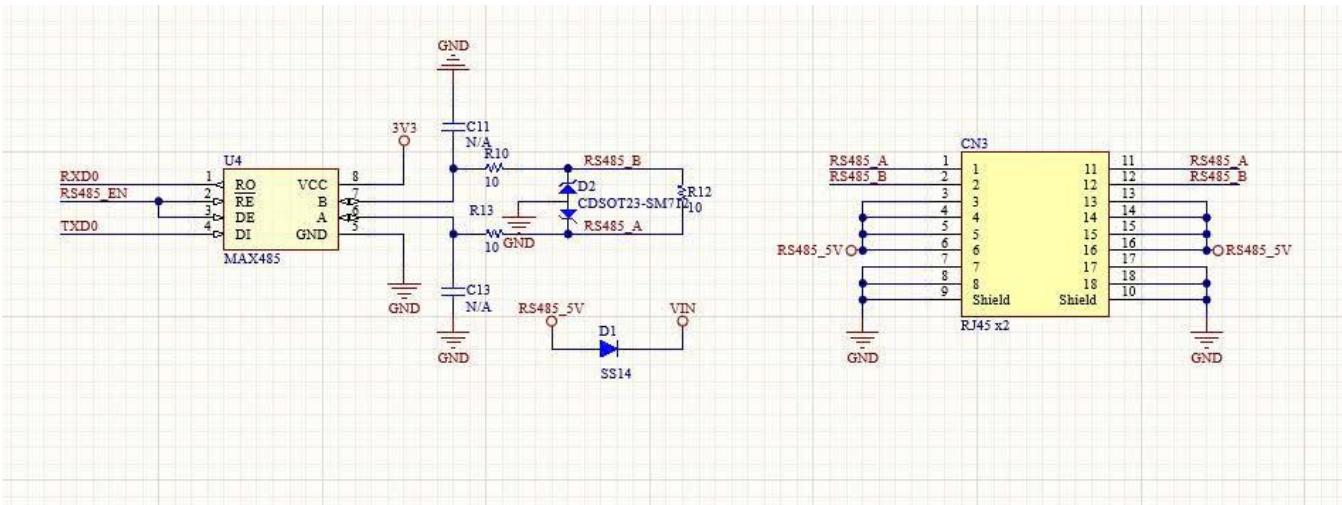


Figure 1.6: Our RS 485

For RS-485 part, we use an IC MAX485 to convert UART signal to 485 signal and vice versa. We also use a RJ45x2 to connect RS-485 signals. Last thing we need to consider for this part is that RJ45x2 is also supply 5V input, so we use D1 to prevent the current go through the RS485_5V pins.

4.6 Interface with high-current LEDs

Now, we design an output part which includes a 2-color LED as shown in Figure 1.7. In this part, we use two transistors to connect with the 2-color LED.

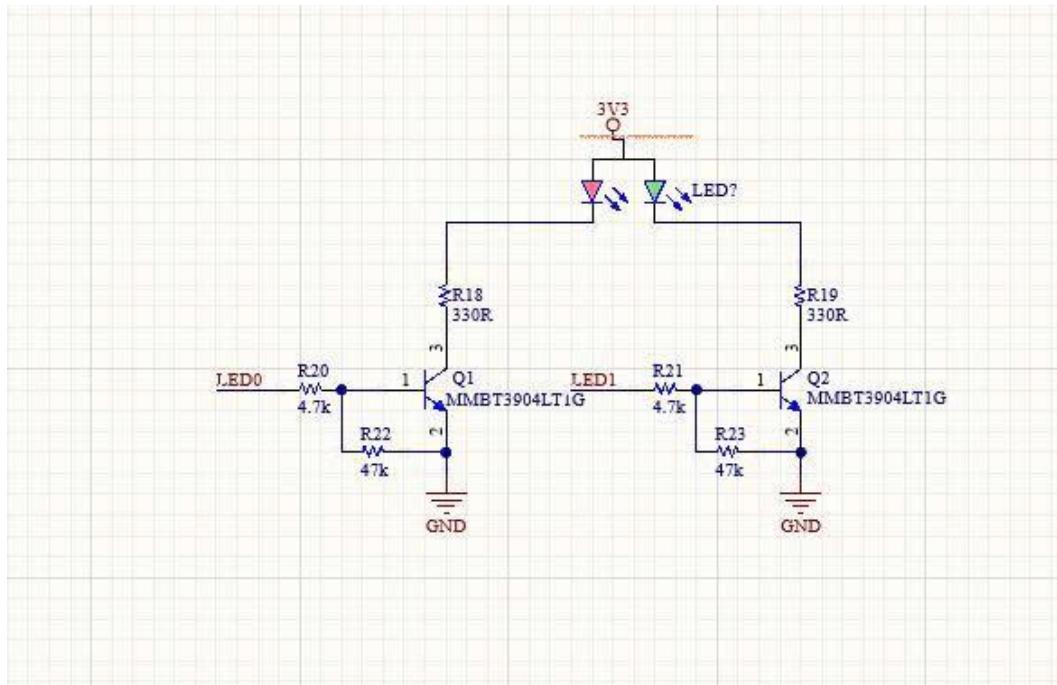
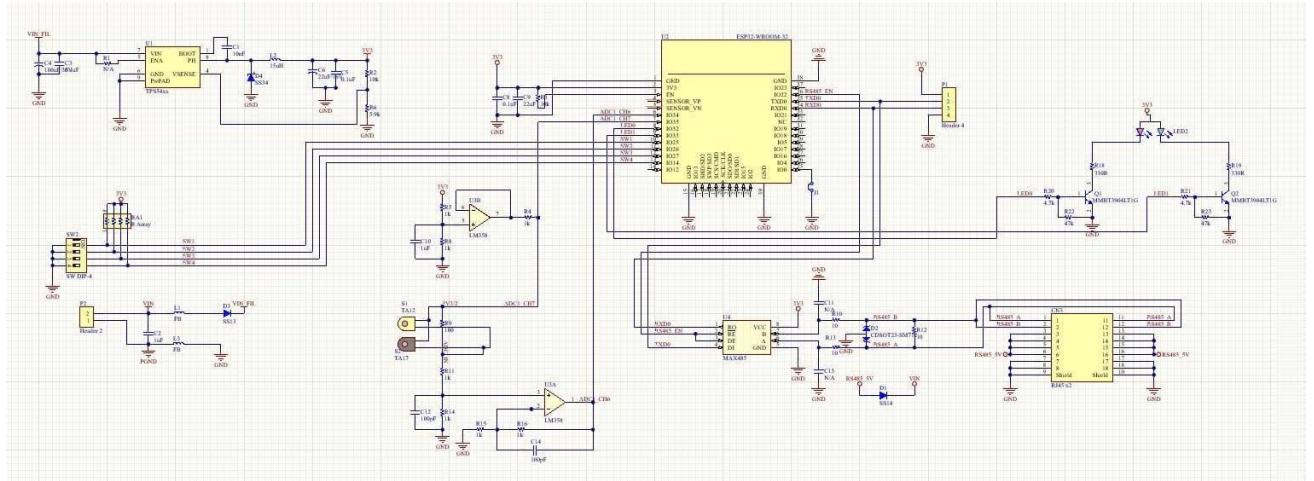


Figure 1.7: Our LED display

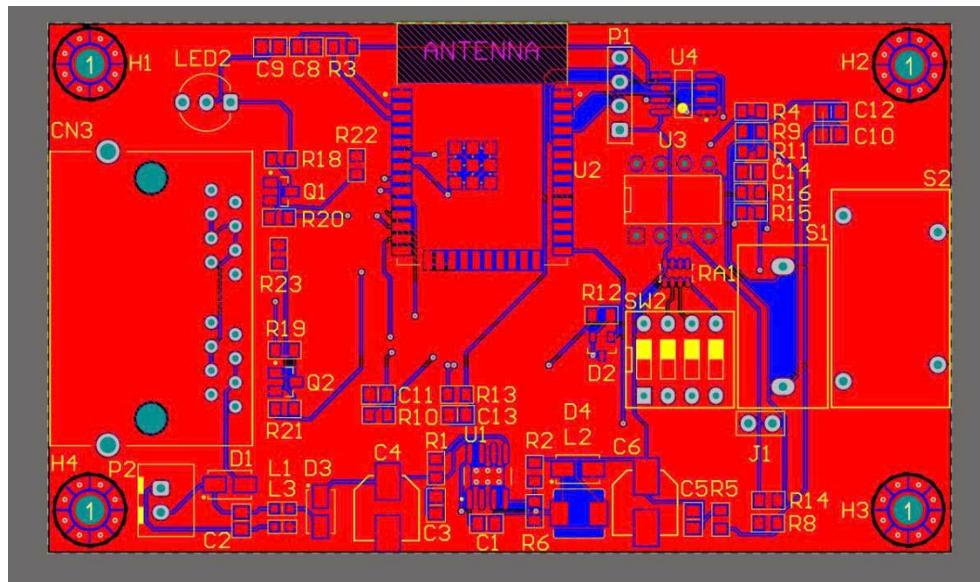
5 Final result

Here is our result of final result:

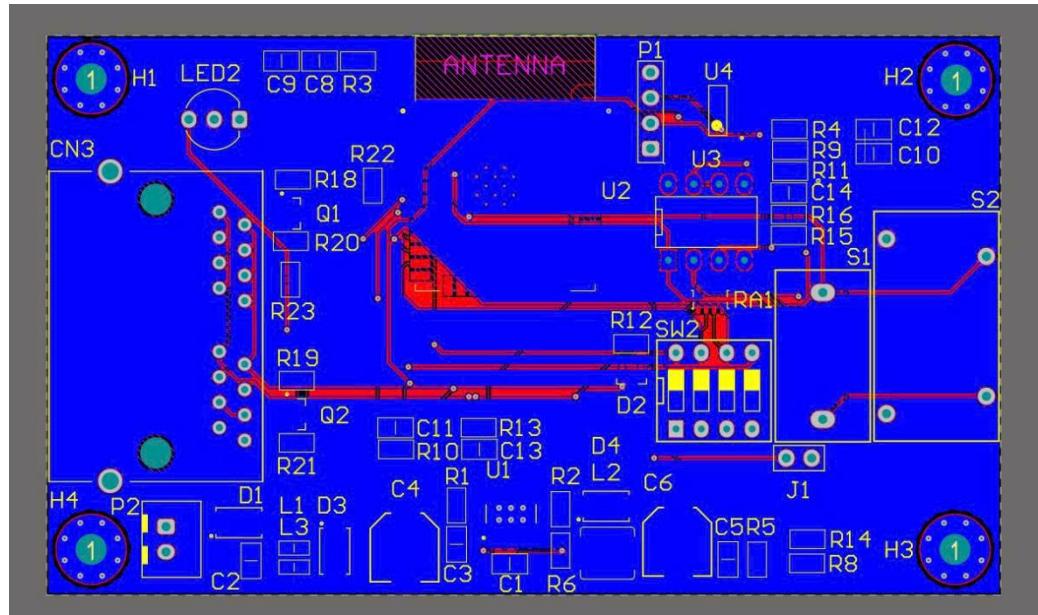


Full Schematic

Top and bottom layer

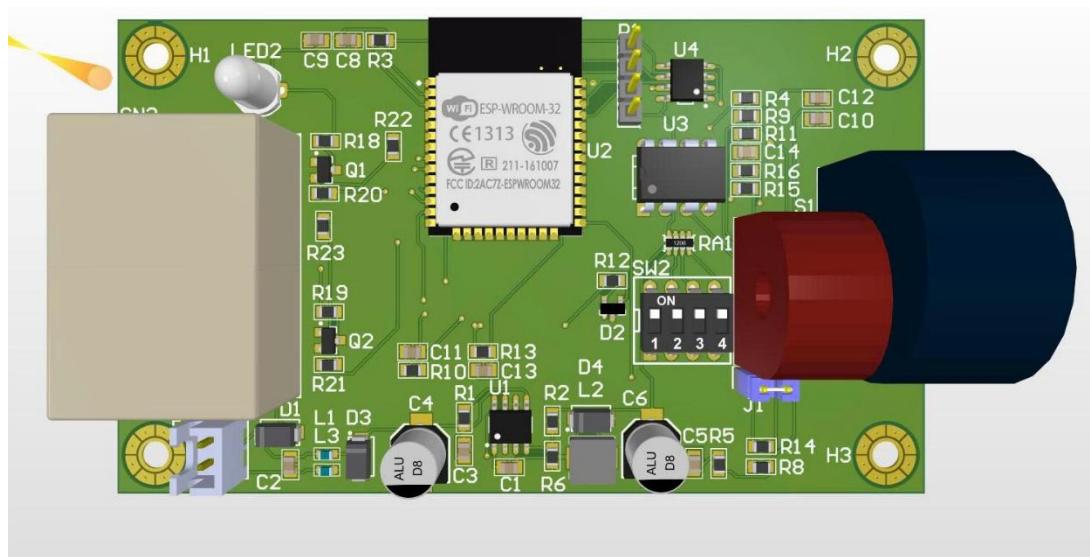


Top layer

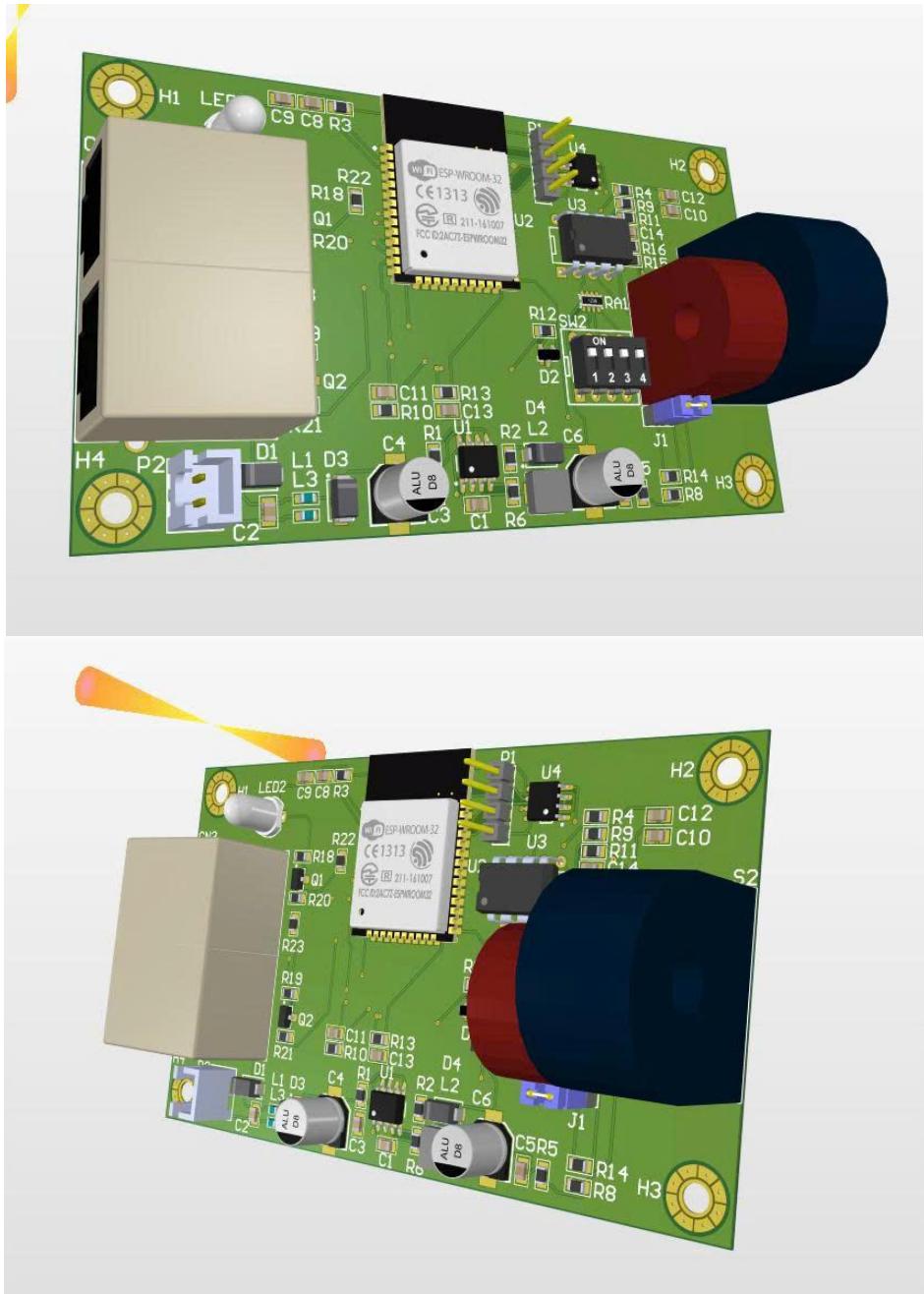


Bottom layer

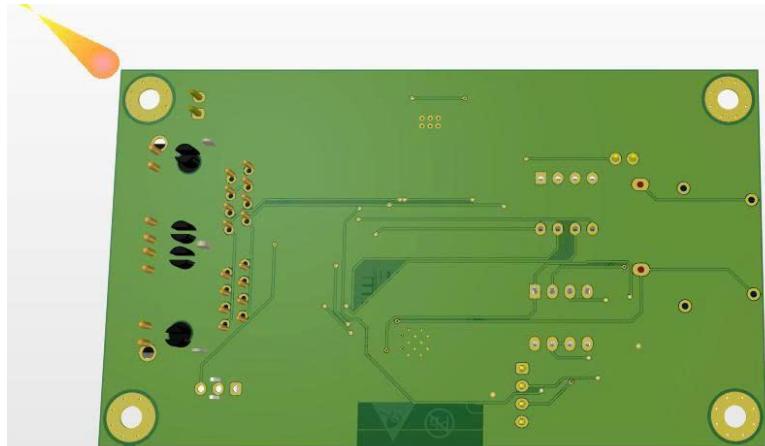
3D View:



Top view of the circuit in 3D model



Side view of the circuit in 3D model



Bottom view of the circuit in 3D model

6 Requirements

6.1 Questions to answer

6.1.1. Research on the Internet and list 5 different current sensors that you can find. Along with each current sensor, please (1) give a reference source, (2) maximum current that the sensor can measure, and (3) how to obtain its values (e.g, using ADC, UART, I2C or SPI and so on):

Answer:

- ACS712 (Hall – Effect Analog Current Sensor)

+ Reference:

<https://www.allegromicro.com/-/media/files/datasheets/acs712-datasheet.pdf>

+ Maximum current:

$\pm 5A$, $\pm 20A$, or $\pm 30A$ (depending on the version of sensor)

+ How to obtain values:

Outputs an analog voltage proportional to the measured current

Read using the MCU ADC

- INA219 (High-Side Current Sensor)

+ Reference:

<https://www.ti.com/lit/ds/symlink/ina219.pdf>

+ Maximum current:

Typically up to $\pm 3.2A$ (depends on shunt value)

+ How to obtain values:

Provides digital measurement over I²C bus.

MCU read voltage, current, and power directly via I²C registers.

- INA226 (High-Precision Current and Power Monitor)

+ Reference:

<https://www.ti.com/lit/ds/symlink/ina226.pdf>

+ Maximum current:

Depends on the external shunt resistor (commonly around 10 – 20 A).

+ How to obtain values:

Communicates using I²C.

Provides digital current, voltage and power readings.

- ACS758 (High-Current Hall-Effect Current Sensor)

+ Reference:

<https://www.allegromicro.com/-/media/files/datasheets/acs758-datasheet.pdf>

+ Max current:

Up to $\pm 50A$, $\pm 100A$, $\pm 150A$, (depending on version)

+ How to obtain values:

Analog output voltage proportional to current.

Read using MCU ADC

- ZMCT102 (AC Current Transformer)

+ Reference:

<https://linhkiendientu.vn/san-pham/cu%E1%BB%99n-d%C3%A2y-ct--zmct102-5a-2.5ma>

+ Maximum current:

5A AC

+ How to obtain values:

The CT outputs a small AC current proportional to the measured current.

A burden resistor converts this current to a small AC voltage.

After biasing/ offset adjustment, the MUC reads the signal through it ADC

6.1.2. In Figure 1.4, what is the voltage of SW1 when slide switch 1 is ON? and is OFF?

Answer:

When slide switch 1 is ON:

The 3.3V power supply is connected directly to GND so the voltage of SW1 is 0V.

When slide switch 1 is OFF:

The 3.3V power supply is connected directly to the SW1 so the voltage of SW1 is: 3.3V

6.1.3. In Figure 1.5, what is the voltage of ADC1_CH7? of ADC1_CH6?

Answer:

We assume no current in TA12, TA17 (no load current), so the current transformers produce no AC signal. Then we only focus on analyse the DC bias of the circuit

Look at the top part of the schematic:

+ R5 = 1k ohm, from 3.3V to the middle node

+ R8 = 1k ohm, from the middle node to GND

+ Middle node connected to the + input of op-amp U3B and U3B is wired as a voltage follower (output fed back to the - input). Then the output of U3B go through R4 (1k ohm) to ADC1_CH7

R5 and R8 form a simple resistor divider:

$$V_{\text{mid}} = V_{\text{in}} \times \frac{R_8}{R_5+R_8} = 3.3 \times \frac{1k}{1k+1k} = 1.65V$$

Because the U3B is a buffer. So it's output (and thus ADC1_CH7) equals this mid voltage

$$\Rightarrow \text{ADC1_CH7} = 1.65V$$

Voltage at ADC1_CH6

The 3V3/2 voltage supply is $3.3/2 = 1.65V$

R9 = 100 ohm 3V3/2 to node ADC_IN

R11 = 1k ohm, from ADC_IN node to the node at the +input of U3A

R14 = 1k ohm, from +input node to GND

U3A is a non-inverting amplifier

R15 = 1k ohm, from intput to GND

R16 = 1k ohm, from ouput back to - input

DC gain:

$$A = 1 + \frac{R_{16}}{R_{15}} = 1 + \frac{1k}{1k} = 2$$

So the ouput (ADC1_CH6) is

$$V_{\text{ADC1_CH6}} = A \cdot V_{+} = 2 \times 0.5V_{\text{ADC_IN}} = V_{\text{ADC_IN}}$$

So the ADC1_CH6 has the same DC voltage as node ADC_IN

6.1.4. In Figure 1.5, we apply a low pass filter to the signal ADC_IN. What is the cutoff frequency of this low pass filter? If we want to set a cutoff frequency is about 10kHz, what should we change in the circuit of U3A?

Answer:

a)

In the circuit around U3A, the low-pass filter is formed by R16 = 1k ohm and C14 = 100 pF.

We use the RC low-pass formula: $f_c = \frac{1}{2\pi R_x C} = \frac{1}{2\pi \times 3.14 \times 1000 \times 100 \times 10^{-12}} = 1.59 \times 10^6 \text{ Hz}$
 \Rightarrow Cutoff frequency = 1.6 MHz

b)

If we want the cutoff frequency = 10kHz

$$C_{\text{new}} = \frac{1}{2\pi R_x f_c} = \frac{1}{2\pi \times 3.14 \times 1000 \times 10000} = 1.6 \times 10^{-8} \text{ F} = 16 \text{ nF}$$

$$R_{\text{new}} = \frac{1}{2\pi f_c C} = 160 \text{ k ohm}$$

We can increase the C14 to 16nF or increase R16 to 160k ohm

6.1.5. How much do the currents go through each LED in Figure 1.7? What should we do if we want to control a 100mWLED?

Answer:

a)

We don't have exact number of Vf of the Red and Green Led. So base on the range of forward voltage of red led (1.7 – 2.0 V) and green led (1.9 – 2.2 V). We assume that the forward voltage of red led is 1.7V and green led is 1.9V.

$$I_{\text{green led}} = \frac{V_{\text{supply}} - V_{\text{ce}} - V_f}{R_{19}} = \frac{3.3 - 0.2 - 1.9}{330} = 3.63 \text{ mA}$$

$$I_{\text{red led}} = \frac{V_{\text{supply}} - V_{\text{ce}} - V_f}{R_{18}} = \frac{3.3 - 0.2 - 1.7}{330} = 4.24 \text{ mA}$$

b)

We are given the LED power:

$$P_{\text{LED}} = 100 \text{ mW} = 0.1 \text{ W}$$

Assume a forward voltage of about:

$$V_F \approx 2 \text{ V}$$

The desired LED current is:

$$I_P = \frac{P_{\text{LED}}}{V_F} = \frac{0.1}{2} = 0.05 \text{ A} = 50 \text{ mA}$$

To get 50 mA from a 3.3 V supply through a transistor (still assuming $V_{CE(sat)} \approx 0.2 \text{ V}$), the new series resistor should be:

$$R_{\text{new}} = \frac{V_s - V_{CE} - V_F}{I_P} = \frac{3.3 - 0.2 - 2.0}{0.05} = \frac{1.1}{0.05} = 22 \Omega$$

So, to control a 100 mW LED:

- + We must decrease the series resistor from 330 Ω to about 22 Ω (a nearby standard value like 22 Ω or 24 Ω is acceptable),
- + And of course ensure that the transistor and resistor can handle the higher current and power dissipation.

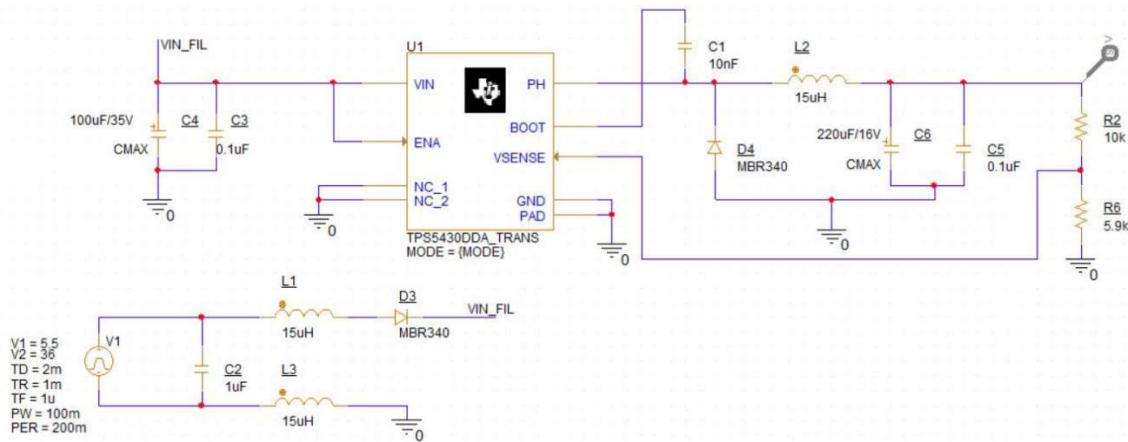
6.1.6. What is the main purpose of D2 in Figure 1.6?

Answer:

D2 9CDSOT23 – SM712) is a TVS protection diode array for the RS-485A and B lines. Its main purpose is to clamp high-voltage transients (ESD, surge, lightning, cable, discharge, etc.) on the RS485_A and RS485_B lines. Also protecting the MAX3485 transceiver and the rest of the circuit from being damaged by those spikes

6.2 PSpice simulation

[1] **3.3V regulator circuit:** Select suitable regulator IC for stable 3.3V output, define in-put voltage range, add output capacitor for noise reduction, simulate transient response, evaluate thermal performance in PSpice.



TPS5430 Switching Regulator: The converter IC operating at fixed 500kHz switching frequency

- Input Voltage Range: 5.5 V to 36 V

- Main function: Convert the input voltage into stable 3.3V output

The circuit has 5 main blocks:

1. Input Filter + Protection

- **V1 (Input Pulse Voltage Source):** Provides a pulsed input voltage to test how the circuit responds to changing or disturbed input conditions.
- **L1, L3, C2 → LC EMI filter to remove noise**
- **D3 (Schottky) → reverse-polarity protection + blocks switching noise**
- Output of this section = **VIN_FIL** (clean input voltage)

2. Input Capacitors

- **C4 (100μF) → bulk input filter**
- **C3 (0.1μF) → high-frequency bypass**
→ They stabilize VIN for the TPS5430.

3. Switching Stage (TPS5430 IC)

- **C1 (10nF) =** Provides a small stored charge that helps the IC switch the PH node properly during operation

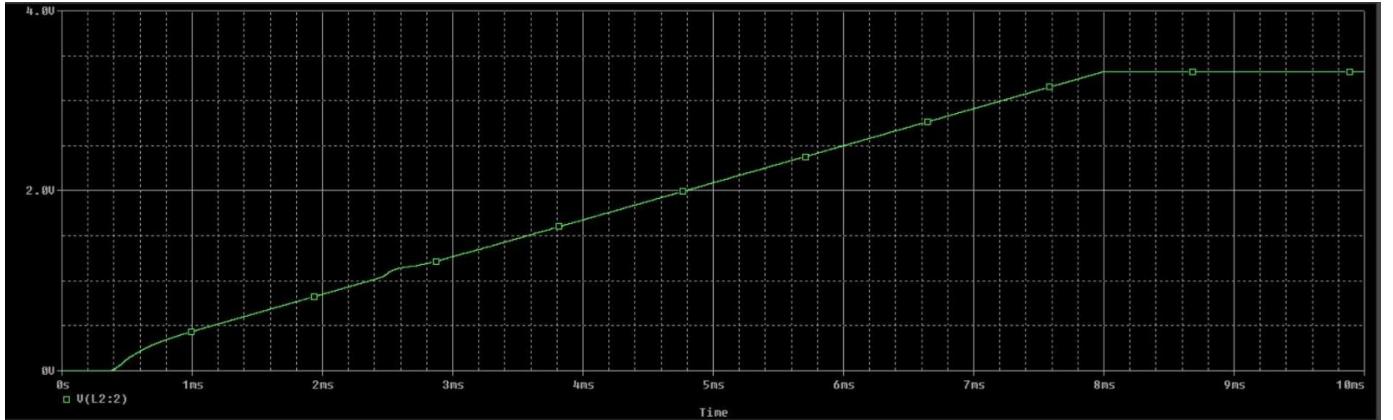
4. Output Buck Filter

- **L2 (15μH) →** smooths current
- **D4 (Schottky) →** freewheel diode
- **C6 (220μF) + C5 (0.1μF) →** reduce ripple and noise
→ Produces a clean DC output.

5. Feedback Divider

- **R2 (10k) + R6 (5.9k)** set the output voltage:

$$V_{out} \approx 3.3V$$



At the beginning of the simulation, the output voltage starts small because the **inductor (L2) and capacitors (C8, C5)** need time to charge. The buck converter cannot produce the final output immediately — the inductor current must ramp up and the output capacitors must accumulate energy.

During this startup period, the control loop inside the **TPS5430** is also stabilizing and adjusting the duty cycle.

That is why the output voltage rises gradually before reaching the steady 3.3 V level.

Thermal Performance Evaluation

Conditions: $V_{IN_FIL} = 36V$, $V_{out} = 3.3V$ (steady state)

We use the Power dissipation (P_{AVG}) to measure the average power dissipation of U1

The Average Power (P_{IC}) IS 74.73 mW. So the total Power Loss is 0.07473

Thermal Calculation and Conclusion

For checking the needs of an external heatsink is required, the Junction Temperature (T_J) is estimated by using thermal resistance formula

$$T_J = T_A + P_{AVG} \cdot \theta_{JA}$$

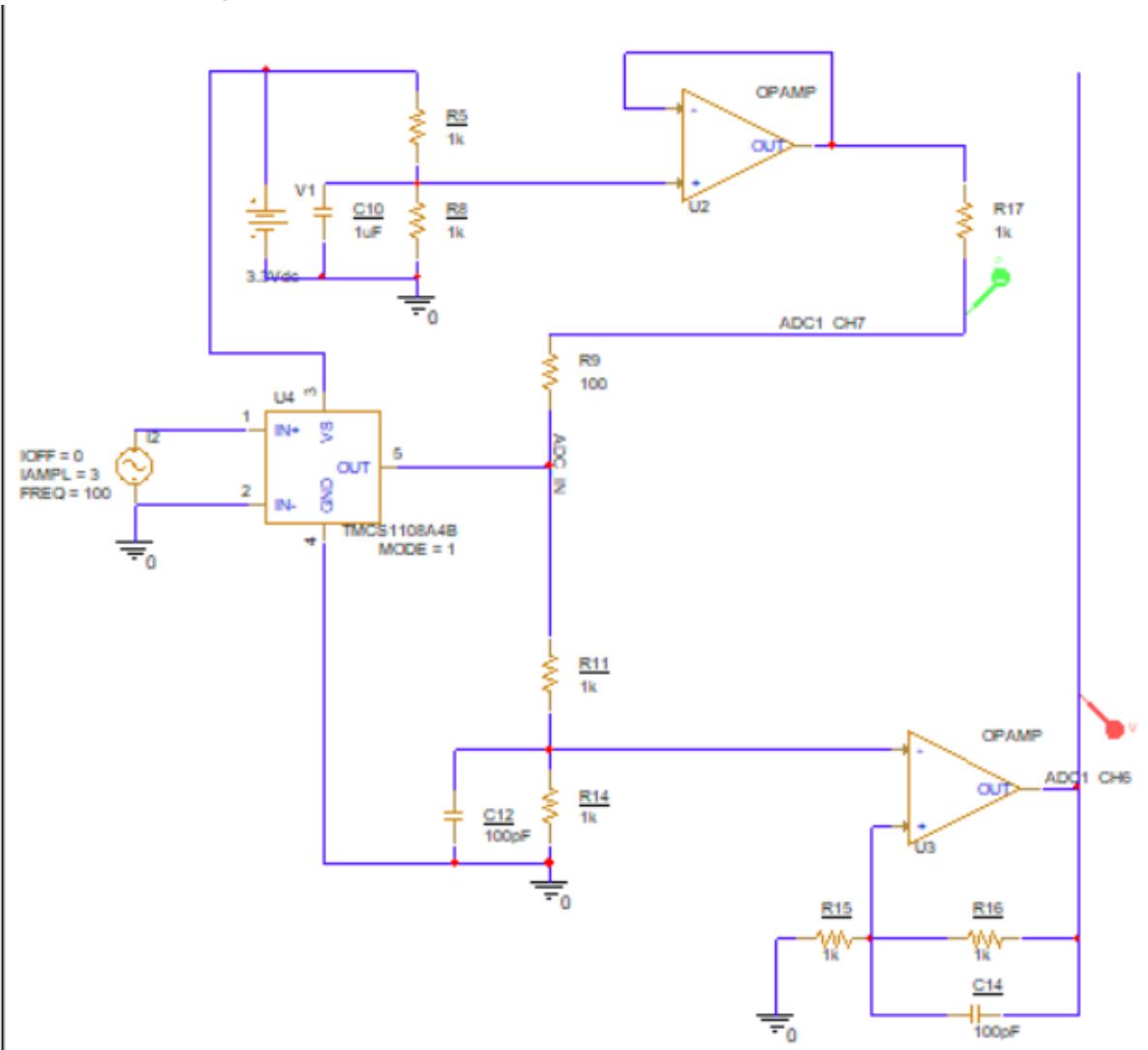
Ambient Temperature (T_A): 25°C

Thermal Resistance (θ_{JA}): Base on the datasheet of TPS5430 we choice $\theta_{JA} = 33^\circ\text{C}$

Total Power (P_{AVG}): 0.0198W

Conclusion: The calculated junction temperature of **27.47 °C** is far below the IC's maximum junction limit of **125 °C**. This indicates that the regulator operates with excellent thermal performance and remains well within safe operating conditions. As a result, the design does not require any external heatsink; standard PCB copper area is sufficient for heat dissipation.

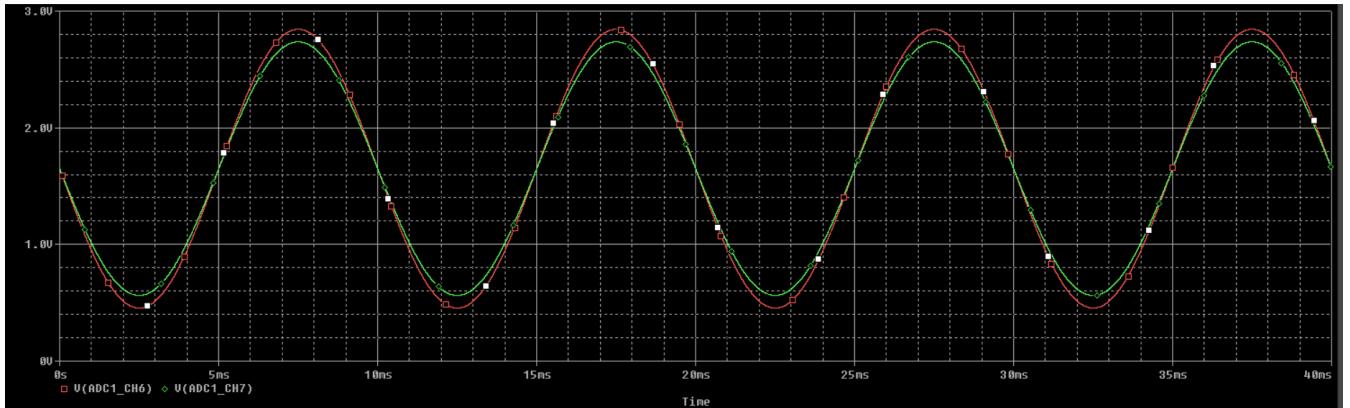
[2] **Current sensor circuit:** Use PSpice sensor for accurate current measurement, set measurement range, add filter to reduce noise and calibrate for minimal error.



Because **TA12 and TA17 current sensors do not exist in PSpice** and have no simulation models, we cannot use them in the simulation environment.

Therefore, we replace them with **TMCS1108A4B** and a current source **ISIN** so the circuit can be simulated correctly.

- TMCS1108A4B is a **Hall-effect current sensor** from Texas Instruments.
- It measures **both AC and DC current** and converts it into a **proportional analog voltage**.
- ISIN is a **PSpice current source** that can generate **DC current, AC sine current, or pulsed current**.
- It is used to simulate the actual current flowing through the sensor input.



The waveform oscillates around **1.65 V** because the TMCS1108A4B is a **Hall-effect current sensor** powered from **3.3 V**.

Its output is designed to sit at **half of the supply voltage (VCC/2)** when the input current is zero.

So:

$$V_{\text{offset}} = \frac{3.3}{2} = 1.65 \text{ V}$$

When current flows, the sensor adds or subtracts a small voltage from this offset:

$$V_{\text{out}} = 1.65 \pm (\text{Sensitivity} \times I)$$

That is why the output is a **sine wave centered at 1.65 V**, not around 0 V.

Filter & Noise Reduction Analysis:

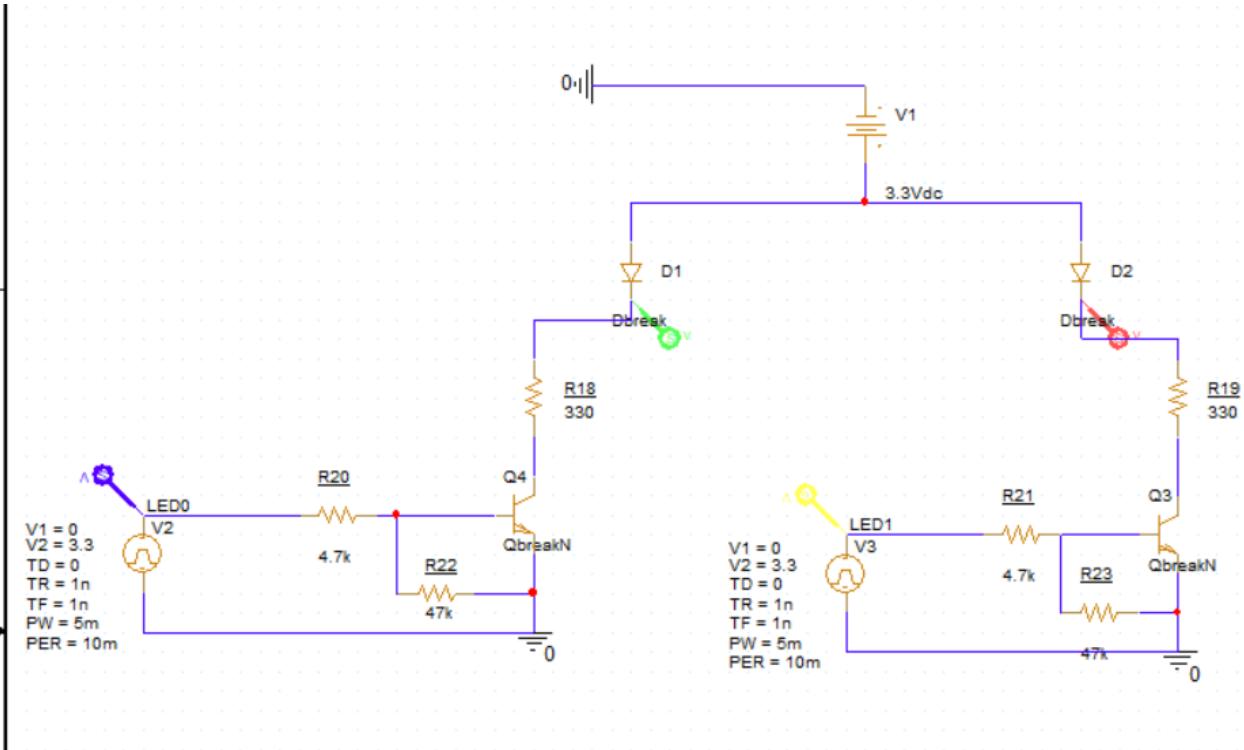
Looking at the simulation schematic, the signal from the sensor does not go directly to the ADC but passes through a Low-Pass Filter (LPF) formed by R11 (1 kΩ) and C12 (100 pF) before entering the Op-amp buffer U3A.

- **Purpose:** This RC filter is critical for removing high-frequency noise and interference from the AC line or switching components (such as the nearby buck converter).
- **Result:** The simulation waveform (red line) shows a smooth sinusoidal wave without jagged edges, demonstrating that the filter effectively cleans the signal before the ADC reads it, fulfilling the design requirement to “reduce noise”.

Measurement Range & Calibration:

- **Sensitivity:** The TMCS1108A4B model used in the simulation typically has a fixed sensitivity (e.g., 100 mV/A).
- **Calibration Verification:** The simulation confirms that at 0 A current, the output sits perfectly at the offset voltage $V_{\text{offset}} = 1.65 \text{ V}$. As the input current swings (controlled by source ISIN), the output voltage swings proportionally within the 3.3 V supply range ($0 \text{ V} < V_{\text{out}} < 3.3 \text{ V}$). This verifies that the circuit is correctly calibrated to measure the target current range (e.g., $\pm 10 \text{ A}$) without clipping or saturation.

[3] **LEDs circuit:** Design LED configuration with current-limiting resistor and simulate PWM for transistor switching.



PWM Theory of Operation:

Pulse Width Modulation (PWM) is a technique utilized to regulate the average power supplied to a load, such as an LED, by rapidly toggling the voltage source between on and off states. The defining characteristic of this method is the Duty Cycle (D), which is expressed by the formula:

$$D = \frac{T_{on}}{T_{period}} \times 100\%$$

In this simulation design, the control signal is configured with a pulse width (T_{on}) of 10ms over a total period (T_{period}) of 20ms. Applying the formula, this results in a 50% Duty Cycle:

$$D = \frac{10ms}{20ms} \times 100\% = 50\%$$

Consequently, the switching frequency is:

$$f = \frac{1}{20ms} = 50Hz$$

At this rate, the transistor switches on and off 50 times per second, effectively reducing the average brightness of the LED to half of its maximum intensity.

Components:

- **V1 (VPulse Source):** Configured to generate the PWM control signal with the timing parameters discussed above (PW = 5 ms, PER = 10 ms).
- **V1 = 0V, V2 = 3.3V:** Defines the logic levels for the transistor switching, matching the 3.3V GPIO voltage of the ESP32.
- **D1 & D2 (Diodes):** Generic diode models (Dbreak) are used in this simulation to represent the LEDs. Note: While generic diodes have a forward voltage ($V_f \approx 0.7 V$), actual LEDs exhibit higher forward

voltage (e.g., Red \approx 1.8 V, Green \approx 2.1 V), which would result in slightly lower current in a real circuit compared to this simulation.

- **R18 & R19 (330 Ω):**

Current-limiting resistors connected in series with the LEDs.

They set the maximum current flowing through the LEDs when the transistors are ON. The value of 330 Ω ensures that the LED current stays within a safe range for standard indicator LEDs.

- **R20 & R21 (4.7 k Ω):**

Base resistors.

They limit the base current flowing from the control source (V1) into the transistors, protecting the MCU's GPIO pins.

- **R22 & R23 (47 k Ω):**

Pull-down resistors.

Connected between the Base and Emitter (GND), they ensure the transistors remain fully OFF (cutoff region) when the control signal is floating or disconnected, preventing unintended switching or noise.

- **Q1 & Q2 (NPN Transistors):**

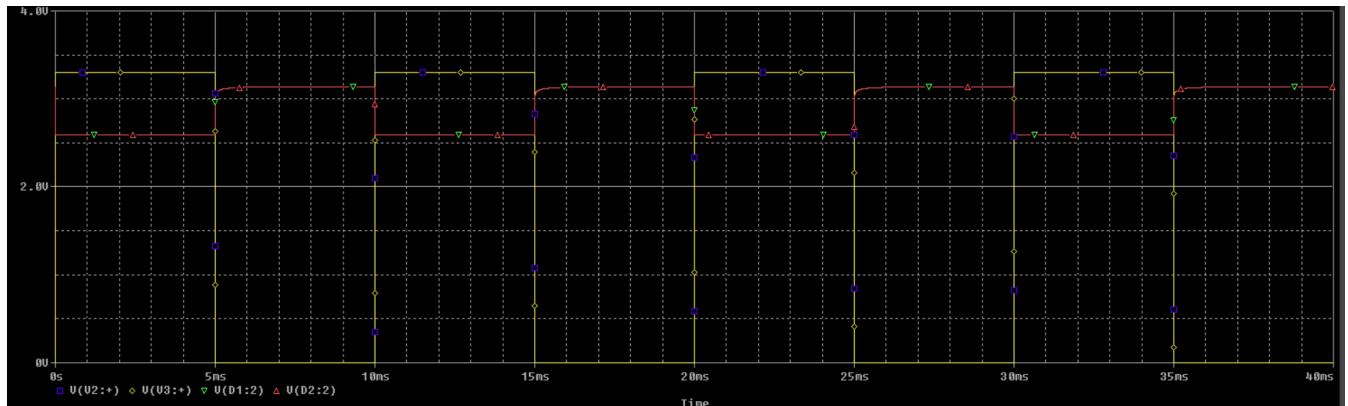
Low-side switches (modeled as QbreakN).

They act as electronic switches controlled by the PWM signal at the Base to drive the LEDs connected to the Collector.

VPULSE is used to simulate the MCU GPIO signal that turns the LED on and off.

- It generates a **0–3.3 V square wave** (PWM-like control signal).
- When VPULSE goes high (3.3 V), the transistor turns **ON** → LED lights up.
- When VPULSE goes low (0 V), the transistor turns **OFF** → LED turns off.

In short: **VPULSE provides the digital control signal for switching the LED in the PSpice simulation.**



These are the **voltages at the LED nodes** (the anodes of D1 and D2).

- When the transistor is **OFF**, the node is pulled near **3.3–5 V** (depends on diode + resistor).
- When the transistor turns **ON**, current flows through the LED → voltage drops to LED forward voltage region.

Red and green curves rise slowly

This is because the LED path includes **resistor + diode**, so the node charges with a small RC delay. That's why you see a **smooth rising curve** instead of an instant step.

When VPULSE drops to 0 V

The transistor turns OFF → the LED current stops → the voltage jumps back to the higher level.

6.3 Requirements of your design and layout

1. Please download this rule <https://github.com/ayberkozgur/jlcpcb-design-rules-stackups>, import it to your Altium Design project. Then you can place and route as

normal.

2. Please note that the name of each component is shown in the figures above. You can use those information to search corresponding components on your project. All the components can be found in the **chipfc_altium_libs**, and also this library https://github.com/chipfc/chipfc_altium_libs

3. Please make the board as small as possible.

7 References

- [1] https://en.wikipedia.org/wiki/Current_sensing
- [2] http://www.electronicoscaldas.com/datasheet/TA12-TA12L-Series_YHDC.pdf
- [3] <https://en.wikipedia.org/wiki/RS-485>
- [4] https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32_datasheet_en.p
- [5] <https://www.ti.com/product/TPS5430>

