Abid Shahriar
Contact: abidshahriar97@gmail.com Alternative: abid047@outlook.com
Mobile : 01677222020

TASK 1:

1. **Project Steps:**

- Defining the Major Steps: The Project involves in making an IOT-Based Embedded System which incorporates some criteria and all these criteria can be followed by the following steps:

1. Selecting Appropriate Micro-Controller Board: Arduino/ESP-32/STM-32 etc. Also, Barebone or Module Based depends on the developer and the project's novelty.

2. PCB Design: Since there are multiple components which is why it is essential to Build a custom-made PCB for the project also selecting 'Through-Hole' or 'SMD-Based Components' plays a major role here.

3. I/O Devices and HMI: we need to first clarify our Input and Output Devices which will also ensure us the communication protocol of them like: I2C, SPI etc.

4. Sensors type Selections and Finalization: Sensors and Sensor's array are crucial part of this project. Depending on the budget and the project's scope we need to select sensors. For an example, in this project we can use Sonic Sensor to measure the depth of the Water that is poured into the glass, right? But to cut the cost we can also try other methods like, let's say a conductive tube will be inserted into its glass and the level of the water will indicate the level as water pours the water level will eventually rise and sections of the tube's conducting part will work as an sensor. Also, if we use 'Threading' function that can also measure the time and we can use time as a meter to adjust the On and Off of the water Tap which will eventually save the cost of a Hardware instead software method can ease the process.

5.M2M: For M2M a MQTT Server can be a good approach for this project.

6. Power Management: We can use different Power source like SMPS, or Linear Voltage Regulators that would eventually serves the purpose of powering small sensors.

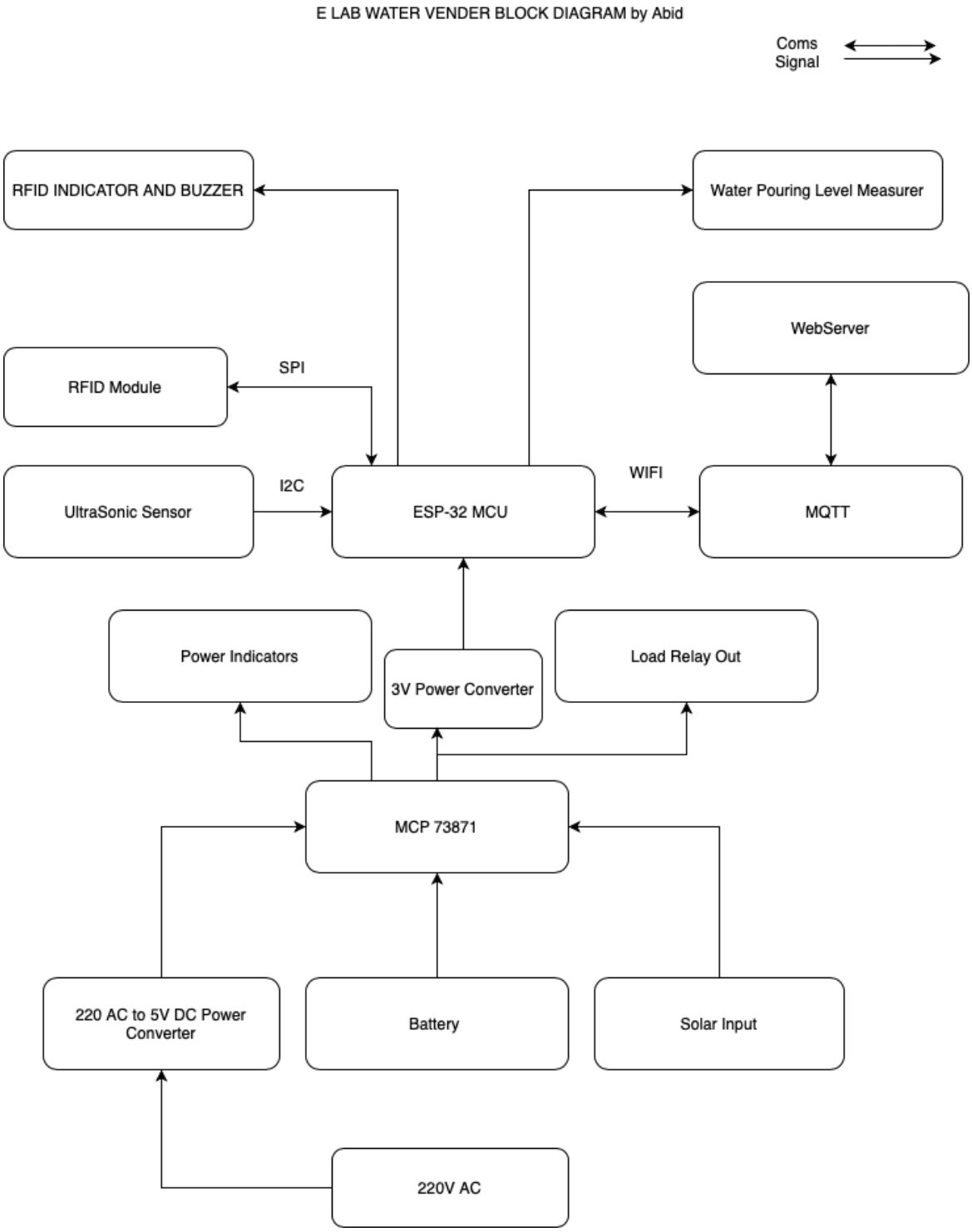
Responsible Parties:
Team Electrical / hardware Engineers: Micro-Controller Selections, PCB Design, Sensors type selections and Finalization, Power Management

Team Firmware Developer: M2M: For M2M MQTT Server development, Appropriate Micro-Controller Board selection and Coding.

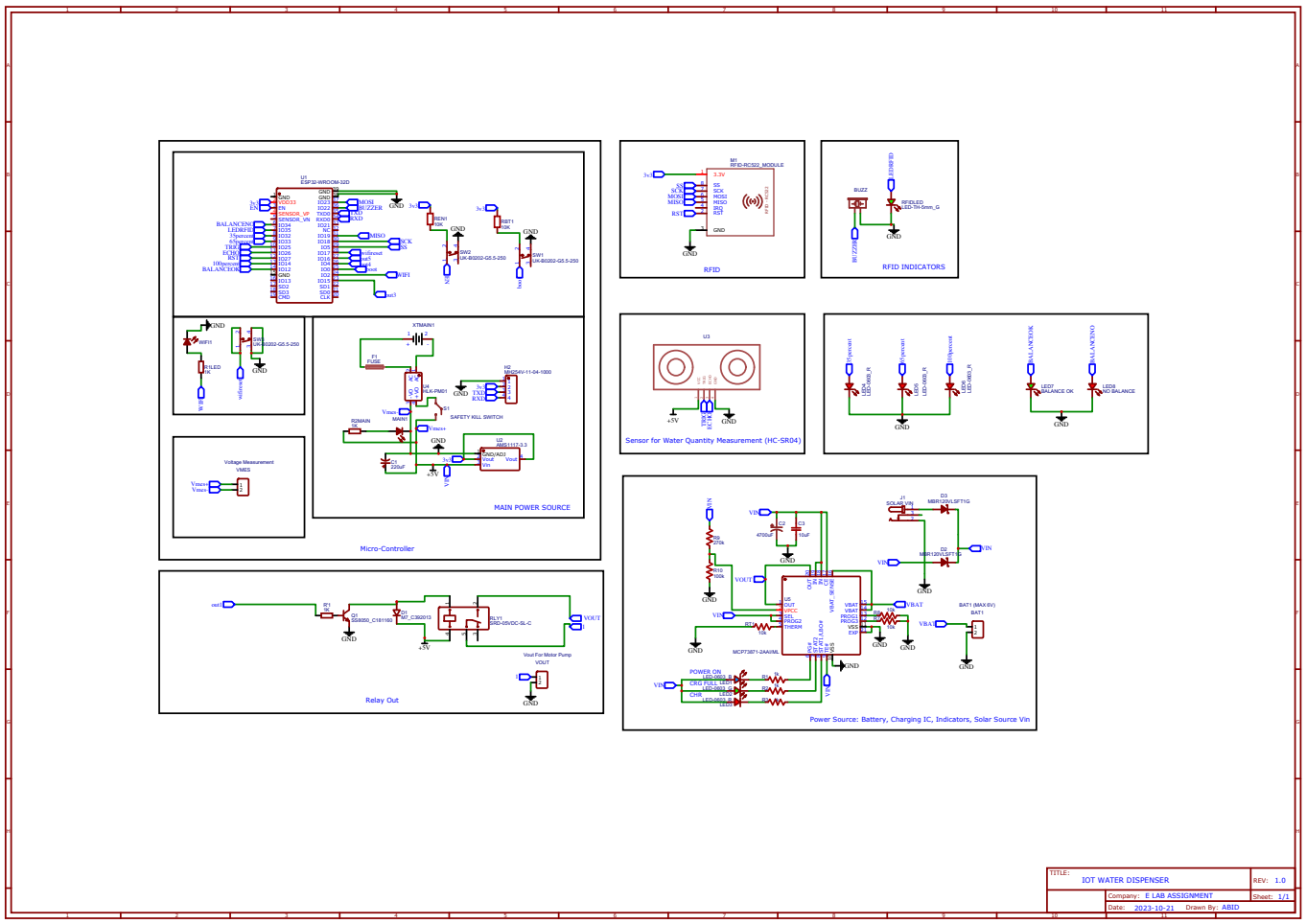
Team Management: This team can better manage the team's Progress and also be crucial for Budget planning.

Team System Reliability / Quality Assurance Engineer: This Team can better handle System Performance and test the system, log datas and find bug and fix errors or can report it to the developer.

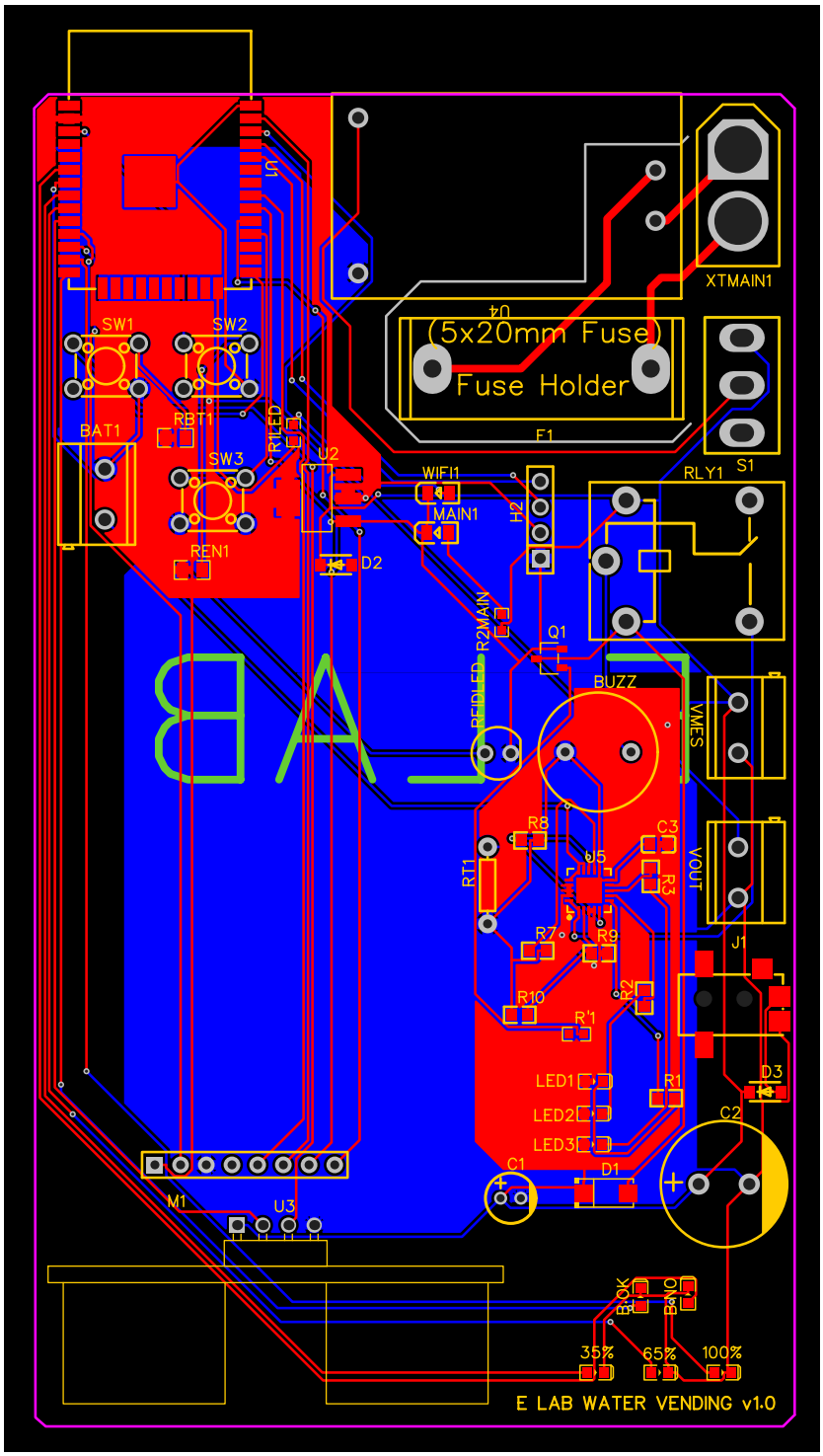
2. Complete the system diagram with the selected component:



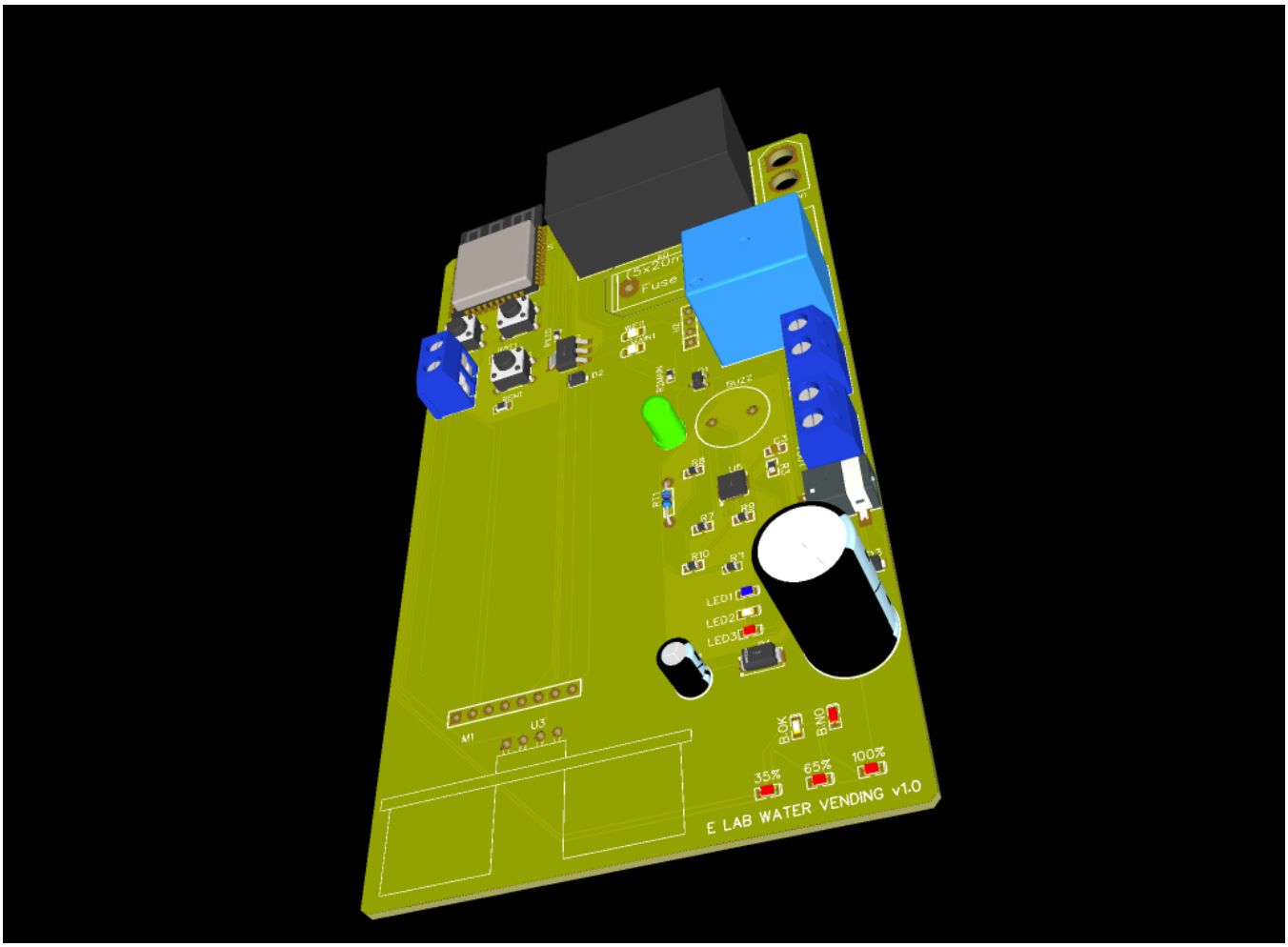
3. High Level hardware design: (For HIRES View: https://github.com/abidshahriar/ELAB-Assignment)



4. PCB Design:


XTMAIN1
(5x20mm Fuse)
Fuse Holder
F1
SW1
SW2
SW3
BAT1
RBT1
REN1
U2
WIFI1
MAIN1
H2
D2
RLY1
S1
R2MAIN
Q1
BUZZ
VMES
VOUT
R8
C3
R3
R7
R9
R10
R'1
J1
LED1
LED2
LED3
R1
D3
C2
C1
D1
M1
U3
BOK
BOO
35%
65%
100%
E LAB WATER VENDING v1.0

3D View:



5.BOM list: (CSV file available here: : https://github.com/abidshahriar/ELAB-Assignment):

ID	Name	Designator	Footprint	Quantity	Manufacturer Part	Manufacturer	Supplier	Supplier Part	Price (USD)
1	BAT1 (MAX 6V)	BAT1	CONN-TH_2P-P5.00	1			LCSC		0.1
2	NO	BUZZ	BUZZER-12MM	1					0.01
3	220uF	C1	CAP-TH_BD5.0-P2.00-D0.8-FD	1	220uF 10V 5*11	ValuePro	LCSC	C43319	0.011
4	4700uF	C2	CAP-TH_BD12.5-P5.00-D1.2-FD	1	EEUFR1A472L	PANASONIC	LCSC	C407873	0.356
5	10uF	C3	C0603	1					0.356
6	M7_C392013	D1	SMA_L4.4-W2.6-LS5.0-RD	1	M7	Guangdong Hottech	LCSC	C392013	0.007
7	MBR120VLSFT1G	"D2,D3"	SOD-123F_L2.8-W1.8-LS3.7-RD	2	MBR120VLSFT1G	ON Semicon	LCSC	C223608	0.135
8	FUSE	F1	FUSE-HOLDER-WITH-COVER-FOR-5X20MM-FUSE_FOOTPRINT	1					0.01
9	MH254V-11-04-1000	H2	HDR-TH_4P-P2.54-V-M-3	1	MH254V-11-04-1000	XFCN	LCSC	C509261	0.096
10	SOLAR VIN	J1	DC-IN-SMD_DC-045B-13A	1	DC-045B-13A HRO		LCSC	C145925	0.183
11	LED-0603_B	LED1	LED0603_BLUE	1	19-217/BHC-ZL1M2RY/3T	EVERLIGHT	LCSC	C72041	0.018
12	LED-0603_G	LED2	LED0603_GREEN	1	19-217/GHC-YR1S2/3T	EVERLIGHT()	LCSC	C72043	0.027
13	LED-0603_R	"LED3,LED4,LED5,LED6"	LED0603_RED	4	19-217/R6C-AL1M2VY/3T	EVERLIGHT	LCSC	C72044	0.016
14	BALANCE OK	LED7	LED0603_GREEN	1	19-217/GHC-YR1S2/3T	EVERLIGHT	LCSC	C72043	0.027
15	NO BALANCE	LED8	LED0603_RED	1	19-217/R6C-AL1M2VY/3T	EVERLIGHT	LCSC	C72044	0.016
16	RFID-RC522_MODULE	M1	RFID-RC522	1					0.35
17	17-21SURC/S530-A3/TR8	"MAIN1,WIFI1"	LED0805-R-RD	2	17-21SURC/S530-A3/TR8	EVERLIGHT	LCSC	C72037	0.018
18	SS8050_C181160	Q1	SOT-23-3_L2.9-W1.6-P1.90-LS2.8-BR	1	SS8050	Hottech	LCSC	C181160	0.01
19	1k	"R1,R2,R3"	R0603	3					0.01
20	1K	"R1LED,R2MAIN,R'1"	R0603	3	RC0603JR-071KL	YAGEO	LCSC	C14676	0.001
21	10k	"R7,R8"	R0603	2					0.5
22	270k	R9	R0603	1					0.5
23	100k	R10	R0603	1					0.5
24	10K	"RBT1,REN1"	R0805	2	0805W8F1002T5E	Uniroyal Elec		LCSC	C38522	0.002
25	LED-TH-5mm_G	RFIDLED	LED-TH_BD5.0_GREEN	1	333-2SYGD/S530-E2	EVERLIGHT	LCSC	C88163	0.034
26	SRD-05VDC-SL-C	RLY1	RELAY-TH_SRD-XXVDC-XL-C	1	SRD-05VDC-SL-C	SONGLE	LCSC	C35449	0.325
27	10k	RT1	R_AXIAL-0.3	1					0.01
28	SAFETY KILL SWITCH	S1	TOGGLE SWITCH SPDT RED 3 PIN ON-OFF-ON	3 SPDT SMALL AC 6A/125V 3A/250V	1				0.05
29	UK-B0202-G5.5-250	"SW1,SW2,SW3"	SW-TH_4P-L6.0-W6.0-P4.50-LS6.5	3	UK-B0202-G5.5-250	USAKRO	LCSC	C620156	0.04

```
30    ESP32-WROOM-32D  U1    WIFIM-SMD_39P-L25.5-W18.0-P1.27-BL 1
      ESP32-WROOM-32D  ESPRESSIF   LCSC  C473012     2.803
31    AMS1117-3.3 U2    SOT-223-3_L6.5-W3.4-P2.30-LS7.0-BR 1     AMS1117-3.3
      AMS   LCSC  C6186 0.141
32    NO    U3    ULTRASONIC-HC-SR04V    1                              3
33    HLK-PM01    U4    PWRM-TH_HLK-PM01 1     HLK-PM01    HI-LINK     LCSC
      C209903     2.299
34    MCP73871-2AAI/ML U5    QFN-20_L4.0-W4.0-P0.50-BL-EP 1     MCP73871-
2AAI/ML     MICROCHIP   LCSC  C144351     5.95
35    Voltage Measurement    VMES  CONN-TH_2P-P5.00 1                 LCSC
      0.3
36    Vout For Motor Pump    VOUT  CONN-TH_2P-P5.00 1                 LCSC
      3
37    XT6   XTMAIN1     XT60 THROUGH HOLE1     X                      0.4

                                                        21.611
                                                        21.61 USD

                                                        2377 BDT
```

1.a. Select major components and Comparision: Answer: Here from the BOM file list we can see the Heart of this project is the ESP32 MCU. The Version We have used Here is `ESP32-WROOM-32D`. This variations of MCU can be used for Reliability and Cutdown cost. Also we have used 220AC to 5VDC conversion components which is `HLK-PM01`. This can also be varied.

Comaprision:

ESP32 WROOM Module	Price	Vendor
ESP32-WROOM-32	$5.95	Digi-Key
ESP32-WROOM-32D	$6.15	Mouser
ESP32-WROOM-32DU	$6.35	SparkFun
ESP32-WROOM-32E	$6.55	Jameco
ESP32-WROOM-32UE	$6.75	eBay

1. b: Why I used ESP32-WROOM-32D:

    a. Its reliable than the other variants
    b. Lesser Dimensions that others
    c. Comparatively Cheaper than others
    d. Just enough GPIO Pins for this Project.
    e. The SRAM is Enough for this project.

5. **Power Management Analysis:**

   a. Discuss power consumption, backup mechanisms, and any power-saving strategies that should be implemented.

So, to discuss with the Power Analysis, I think if we can show a Power Network, then that will be better to analyze.

5V Power Network:

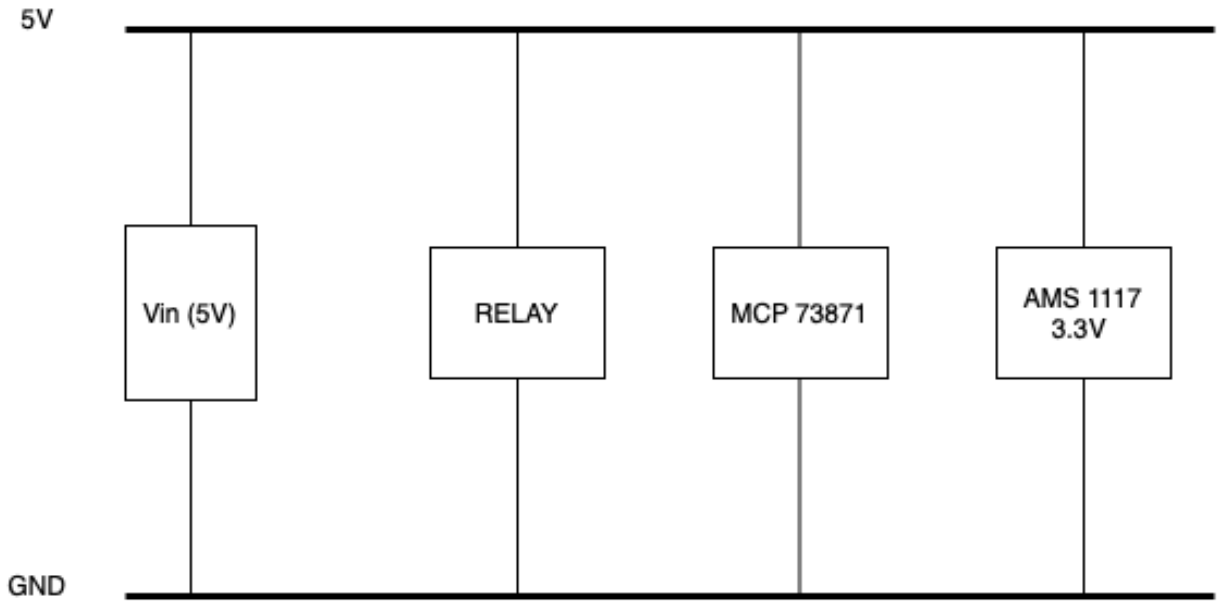


Figure X

3.3V Power Network:

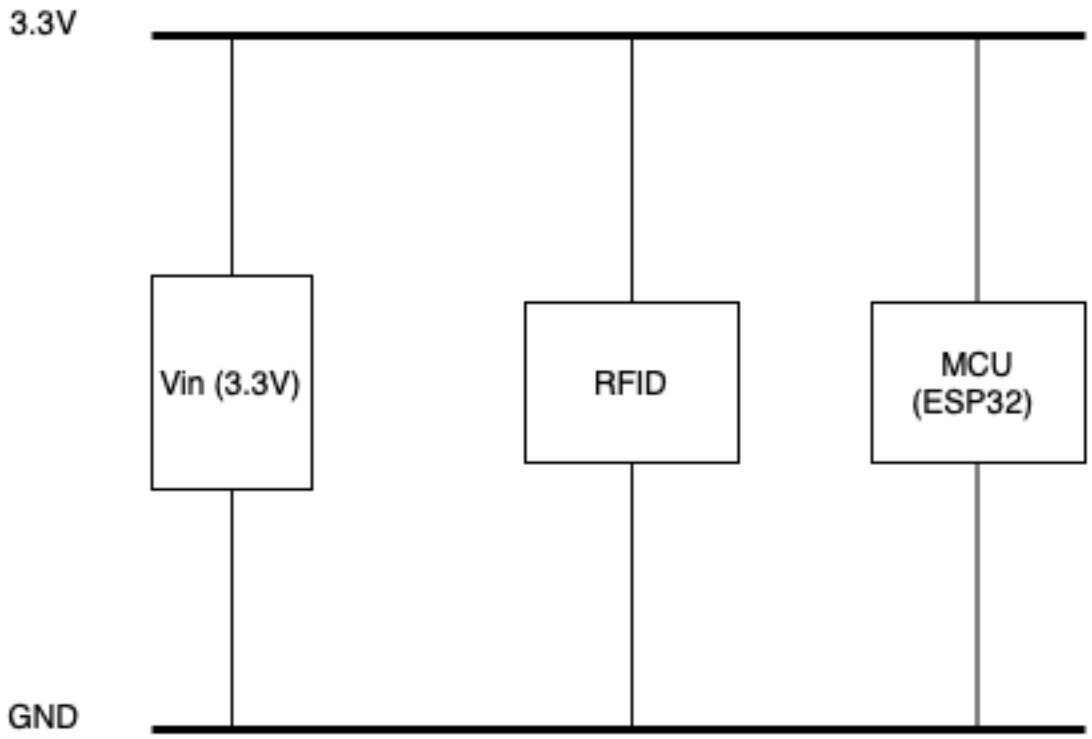


Figure Y

Description and Analysis: Power Network analysis is the best way for any power management related concern of an embedded system. In this system, We can see that the Relay and the loads are connected in parallel to the 5V network. Now the following calculations:

Name	Voltage	Current	Power
Relay	5V	20mA	0.1 W
MCP 73871	5V	100mA	0.5W
AMS 1117	5V	5mA	0.025W

For 3V Network:

Name	Voltage	Current	Power
ESP32	3.3V	270mA	0.9W
RFID	3.3V	200mA	0.7W

So, as we can see the main components that takes very low power consumption. Also For backup system there is Battery and Solar input Vin with the Adapter Module. So this system is reliable.

Next, if we want to run it in really really low power then we can code the MCU to run in low power mode, also ESP32 supports deep sleep mode.

8. Documentation:

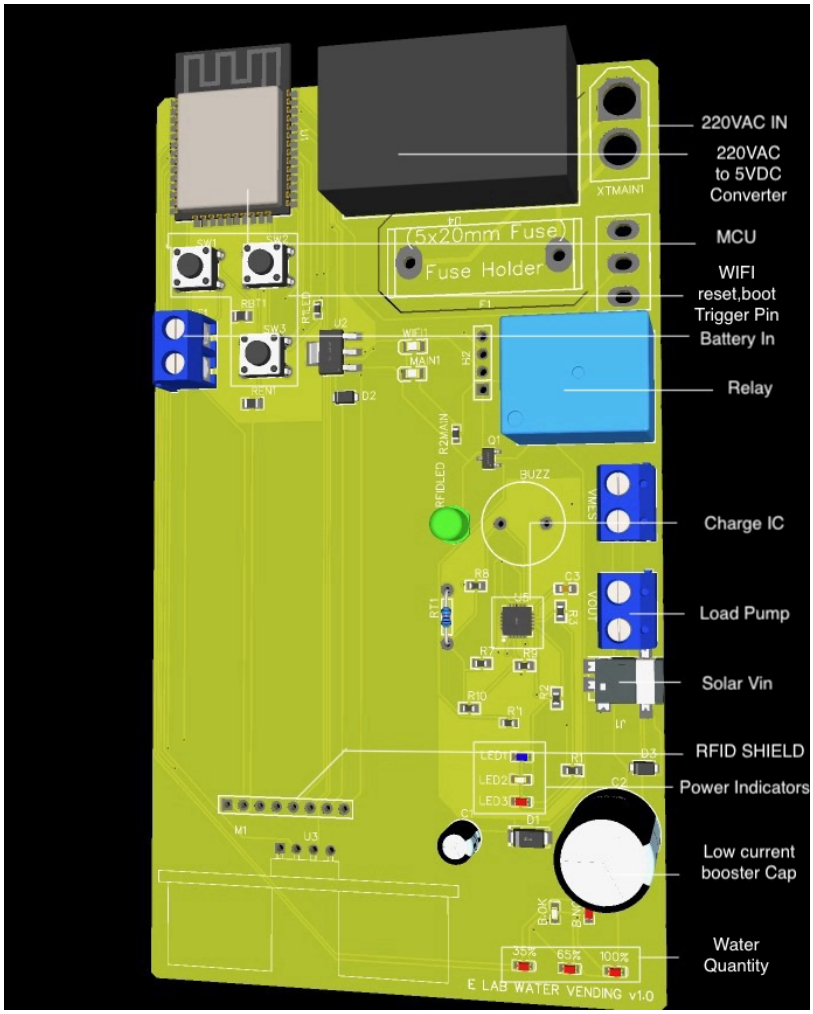


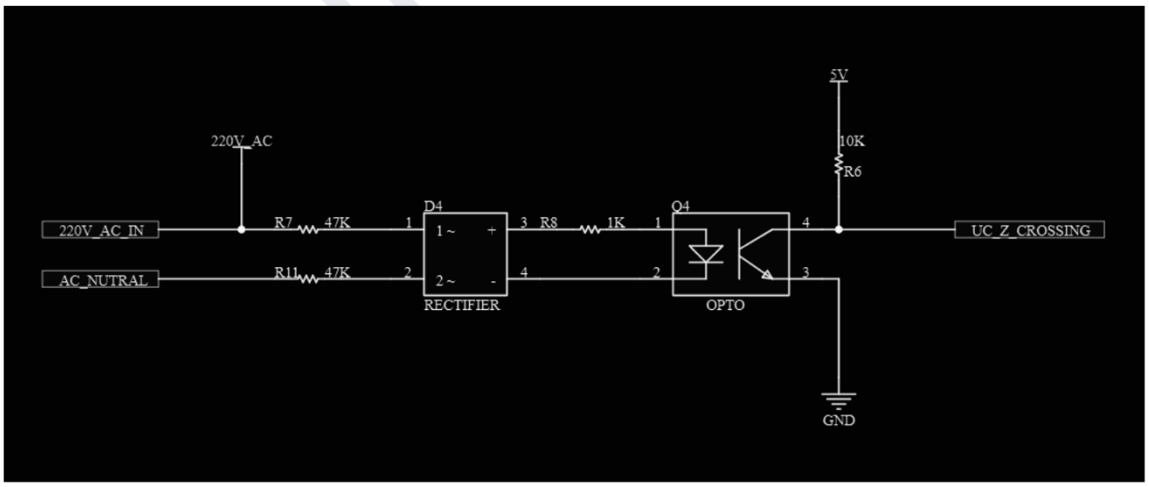
Figure B: Circuit with Marks

The Figure B is itself Self-explanatory with all the Markings of the inbuilt peripherals. In the circuit I have considered 3 Power Source: 1. Max 6V Li-Po Battery Vin, 2: Solar Vin, 3. Power Adapter Vin. **The Special Consideration of this project** is this Multiple Source of Power supply which was the project requirements given and to comply this I have used MCP73871-2AAI/ML IC which have the potential to charge the battery and take two Voltage Input also provides power indicators. The Big capacitor is lined with the Solar Power Supply that helps to boost the current flow when the solar supply may be inefficient due to weather conditions. Sonar Sensor is used to measure the quantity of the water pouring and RFID Shield for the authentication.

The Limitation of the project is it has been designed with very lower power consumption peripherals hence on the load side it can drive a 5V load DC water motor pump. Beyond that capability the system will not be able to provide.

To, Improve this Booster circuit can be used or The pump should be used with amn external power Supply.

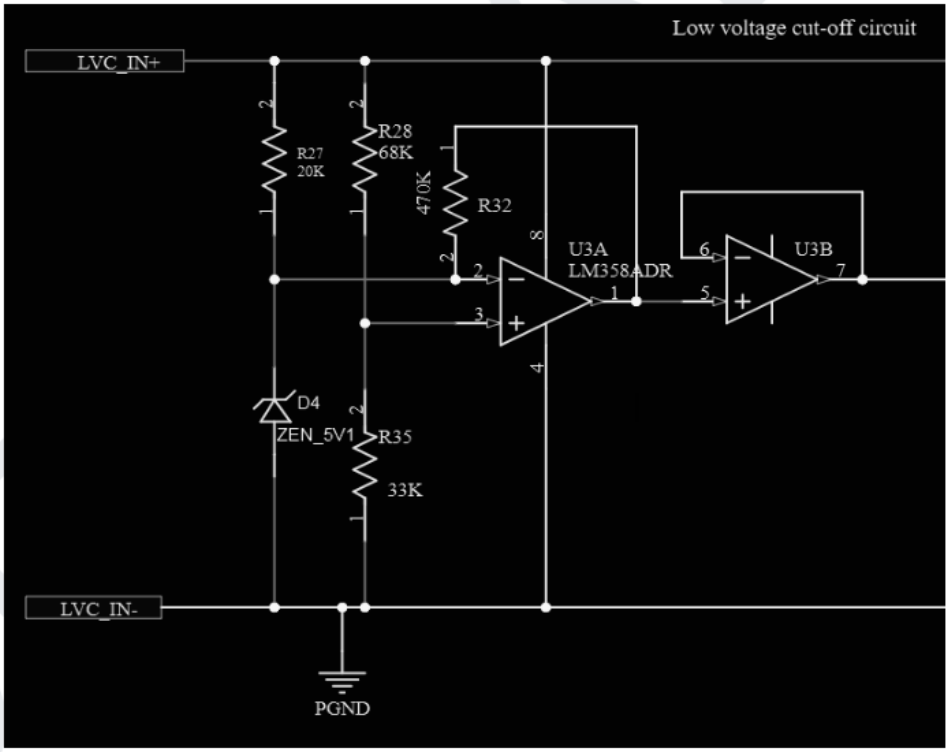
3. Circuit 3:



Analysis: This circuit is an Optocoupler Circuit Basic Diagram. This type of circuit is very useful to isolate High Tension Side voltage with low tension Side Voltage. For an example, Let's say in a Home Automation PCB, let's Say we shall need a Relay to operate. Now to interface with The Relay can get Signal from the Microcontroller and on the load side of the Relay let's say we want to put an AC Load (220V AC). Now it is always a good practice to isolate this High-Tension Side voltage unless we are using any PCB Cutout as for the sake of safety so in any case the high side voltage even in any case goes to the low side that won't affect the micro controller in anyway.

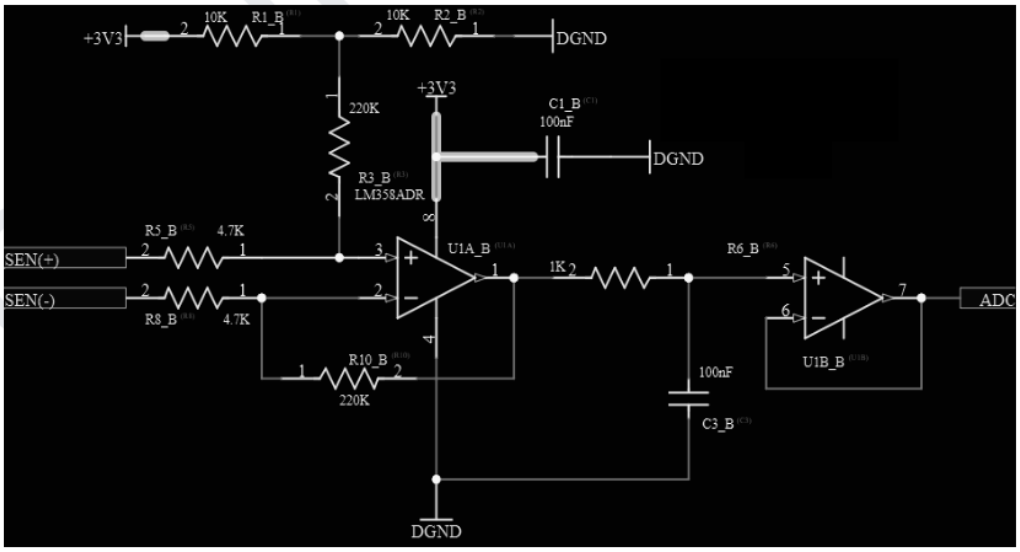
However, here the functionality of the overall circuit works as a zero-crossing detector. As on the left side of the circuit there is AC voltage so the waveform would be Sinusoidal. So, the circuit would detect zero crossing meaning the output wave would be high when the input AC wave crosses zero each time after every 180-degree cycle there would be a peak in the output.

2. Circuit 2:



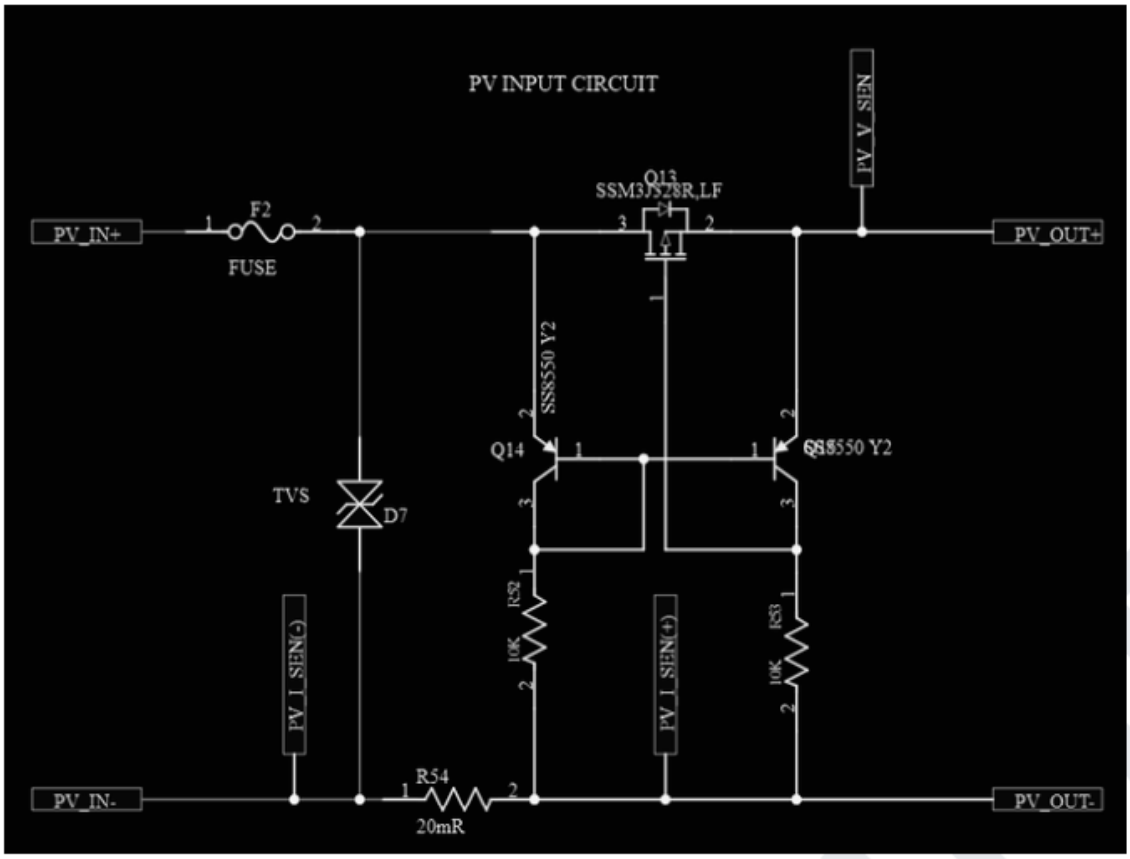
2. Answer: This Circuit uses LM358ADR which has two op-amps. One here is used as voltage comparator. If we notice the two resistors here are used for voltage divider to setup the reference voltage and the one with Zener diode is used for protection. The voltage divider here used to monitor the fraction of the voltage. The first op-amp is used here as a comparator and the other ones is connected with the output to the first one. Now the op-amp will compare the input voltage with the reference voltage and when the input voltage falls below the reference voltage then it will change the state. This can be added with a relay to cut off power of the load when the voltage is low.

5. Circuit 5:



5.Answer: I think this circuit is a amplifier circuit. The first Op-Amp's Vcc to the positive supply voltage and DGND(Ground) to the reference. The Inverting input of the LM358(-) is connected To the output with a resistor as a feedback and also this output is connected with the other ones input signal. I think the second opamp is used to amplify the other signal SEN(-) and then all these signals are send to ADC(Analog to Digital Converter)

1. Circuit 1:



1.Answer: Here main components are one P-Channel Mosfet and two ss8550 transistors.

Task 4: Descriptive Questions

1. Describe a specific challenge you encountered in a past project related to IoT systems in embedded hardware design. How did you diagnose the issue, and what steps did you take to resolve it?

Answer:
In my past IOT based project which I used most of the components as SMD components rather than through hole which was a challenge for me. As Small Pcb which is around 100mmx100mm in Size some of these PCB can be fabricated in very cheap like 5 borads can take just 2-4USD. So, recently I have designed a IOT based home automation board where using SMD components I was able to design it Successfully. Some crucial steps I have also taken was power budget calculation.

2. Can you share an example from your experience where you had to manage an embedded system project with tight deadlines and resource constraints? How did you ensure timely delivery and quality, and how did you coordinate with your team under pressure?

Answer: Yes, I have worked in a such Challenging Environment. I had the chance to work with University Rover Challenge Competition where we had to remake a SAR video with everything unassembled. To ensure this we divided our team into Subgroups and worked straight 18 hours to accomplish it. We had battery constraints as some batteries ran out of charge, so we as electrical team used booster circuits to get even with low quantity battery. Again we have done to etching and make our own PCB.

3. How do you keep your embedded hardware design skills current with evolving technologies and industry trends? Can you provide an example where you applied new knowledge or a technique to a project that you learned from continuous professional development?

Answer: So recently , I have to analyze a circuit board before sending it to fabrication. As I am gradually switching to Altium, I recently came to know about a tools which is about PDN analyzer. This tools actually gives a developer to analyze the Voltage drop and Current Density Analysis across the circuit board. The Boards Simulation shows Heatmaps as voltage drops and these heatmaps can be better analyzed to ensure efficient PCB design like : Providing more efficient Return Path so the trace or VIAS don’t burn out due to higher current density also Impedance matching.

4 Explain a situation where you had to transfer your whole design to a different platform (e.g., Eagle to KiCad). What challenges did you face? How did you mitigate them? What steps would you take next time?

Answer: Yes, I like to do most of the PCB designs in EASYEDA but I am Switching to Altium and Altium is really a great PCB Designing Software. As much as I am really into EasyEDA but it comes with much limitations. In a past project, I had more components in PCB design and Routing Manually was too much challenging. The direction of one components was mistakenly not correct and the EASYEDA was crashing as there was too much trace lines. So, I switched my entire design to Altium and amazingly for the First time I have learned a new feature of Altium which is kind of adaptive tracing. In short: when trace line we try to reroute other nearby trace lines automatically adjusts so the routing becomes easy and very prominent.

5. Considering the IoT product development industry, describe a scenario where you had to design embedded hardware adhering to the specific standards and regulations of that industry. How did you ensure compliance and manage the associated challenges?

Answer: So, I think this question mainly involves with IEEE standards and protocol and also when considering a project safety protocols and guideline with ethical concerns.The following are some of the IEEE standards :

IEEE 802.11 (Wi-Fi):

IEEE 802.11 standards specify wireless LAN (Wi-Fi) technology, including different versions like 802.11a, 802.11b, 802.11n, 802.11ac, and others.
IEEE 802.3 (Ethernet):

These standards cover Ethernet technology, including the widely used 802.3ab (Gigabit Ethernet) and 802.3ae (10 Gigabit Ethernet).

Relative SDG standards:

SDG 7: Affordable and Clean Energy:

The International Renewable Energy Agency (IRENA) offers guidelines and standards for renewable energy sources and energy efficiency.

Code Shipsets:

I do not have very comprehensive knowledge on Coding but I try to learn. Please excuse me for any mistake. For better view : [https://github.com/abidshahriar/ELAB-Assignment)](https://github.com/abidshahriar/ELAB-Assignment)

Ultrasonic Sensor Code:

```
//For the ultra soniuc sensor Measurement

const int trigPin = 25;
const int echoPin = 26;
const int led100 = 14;
const int led35 = 32;
const int led65 = 33;

void setup() {
  Serial.begin(9600);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(led100, OUTPUT);
  pinMode(led35, OUTPUT);
  pinMode(led65, OUTPUT);
}

void loop() {
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  long duration = pulseIn(echoPin, HIGH);
  float distance = duration / 58.2;

  if (distance <= 33) {
    digitalWrite(led100, HIGH);
    digitalWrite(led35, LOW);
    digitalWrite(led65, LOW);
  } else if (distance <= 66) {
    digitalWrite(led100, LOW);
    digitalWrite(led35, HIGH);
    digitalWrite(led65, LOW);
  } else {
    digitalWrite(led100, LOW);
    digitalWrite(led35, LOW);
    digitalWrite(led65, HIGH);
  }

  Serial.print("Distance: ");
  Serial.print(distance);
  Serial.println(" cm");

  delay(1000);
}
```

MQTT Server and WIFI Setup Code:

```
#include <ESP8266WiFi.h>

const char* ssid = "WiFi";
const char* password = "Password";
const char* mqttServer = "MQTTBroker";
const int mqttPort = 0 ;

WiFiClient espClient;

void setup() {
  Serial.begin(9600);
  WiFi.begin(ssid, password);
  client.setServer(mqttServer, mqttPort);
}

void loop() {
  if (!client.connected()) {
    reconnect();
  }
  client.loop();

}

void reconnect() {
  while (!client.connected()) {
    if (client.connect("ESP8266Client")) {

    } else {
      delay(5000);
    }
  }
}
```

For RFID:

```
#include <SPI.h>
#include <MFRC522.h>

#define RST_PIN 5
#define SS_PIN 27
#define LED_PIN 35
#define BUZZER_PIN 22

MFRC522 mfrc522(SS_PIN, RST_PIN);

void setup() {
  Serial.begin(115200);
  SPI.begin();
  mfrc522.PCD_Init();
  pinMode(LED_PIN, OUTPUT);
  pinMode(BUZZER_PIN, OUTPUT);
  Serial.println("OK...");
}

void loop() {
  digitalWrite(LED_PIN, LOW);
  digitalWrite(BUZZER_PIN, LOW);

  if (mfrc522.PICC_IsNewCardPresent() && mfrc522.PICC_ReadCardSerial()) {
    Serial.println("Card detected");
    digitalWrite(LED_PIN, HIGH);
    digitalWrite(BUZZER_PIN, HIGH);
    delay(1000);
    digitalWrite(LED_PIN, LOW);
    digitalWrite(BUZZER_PIN, LOW);
    mfrc522.PICC_HaltA();
  }
}
```

This is the End. I have taken much time and put efforts to complete this. Please Excuse me for any Mistakes I have done. Thanks.