TEAM KNK 2.0
PSG COLLEGE OF TECHNOLOGY, COIMBATORE

Team Members : Aravind T
Dinesh B
Sujithkumar M A
Vaishnavee K R

Idea - Minimizing energy wastage in household appliances using Internet of Things and Arduino UNO

Abstract - Energy Efficient Home

❖ **Water Heater Control**

Getting Temperature Data via TMP36 Sensor. Alert Buzzer and notification message to the user through WiFi

❖ **Water Transferring Motor Control**

Water level is detected using water level detector. No water - Motor turn off - Informed to user through WiFi
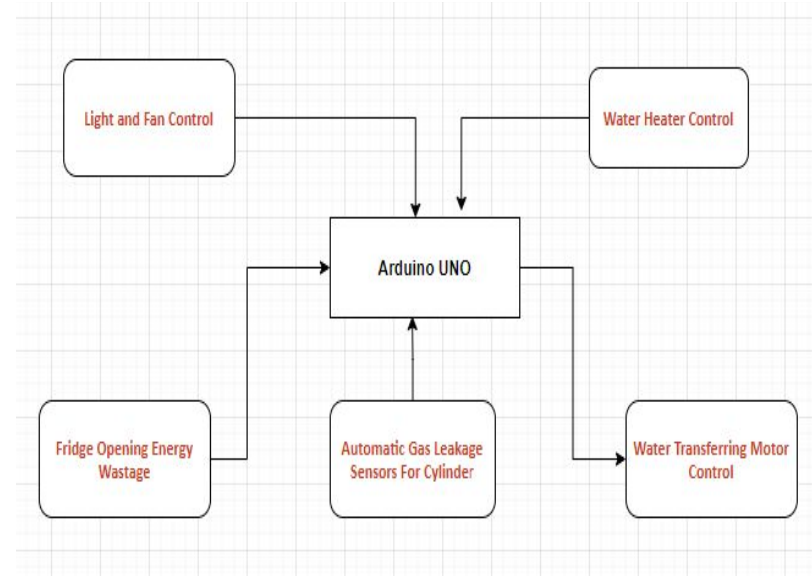
❖ **Automatic Gas Leakage Sensors For Cylinder**

A Gas Sensor - Fitted with cylinder. If gas leaks, user is notified through buzzer and through text via WiFi

❖ **Fridge Opening Energy Wastage**

If fridge opened for a long, energy is wasted. Low cost fridges doesn't have automatic alert. So, alert is provided using Temperature sensor and buzzer.
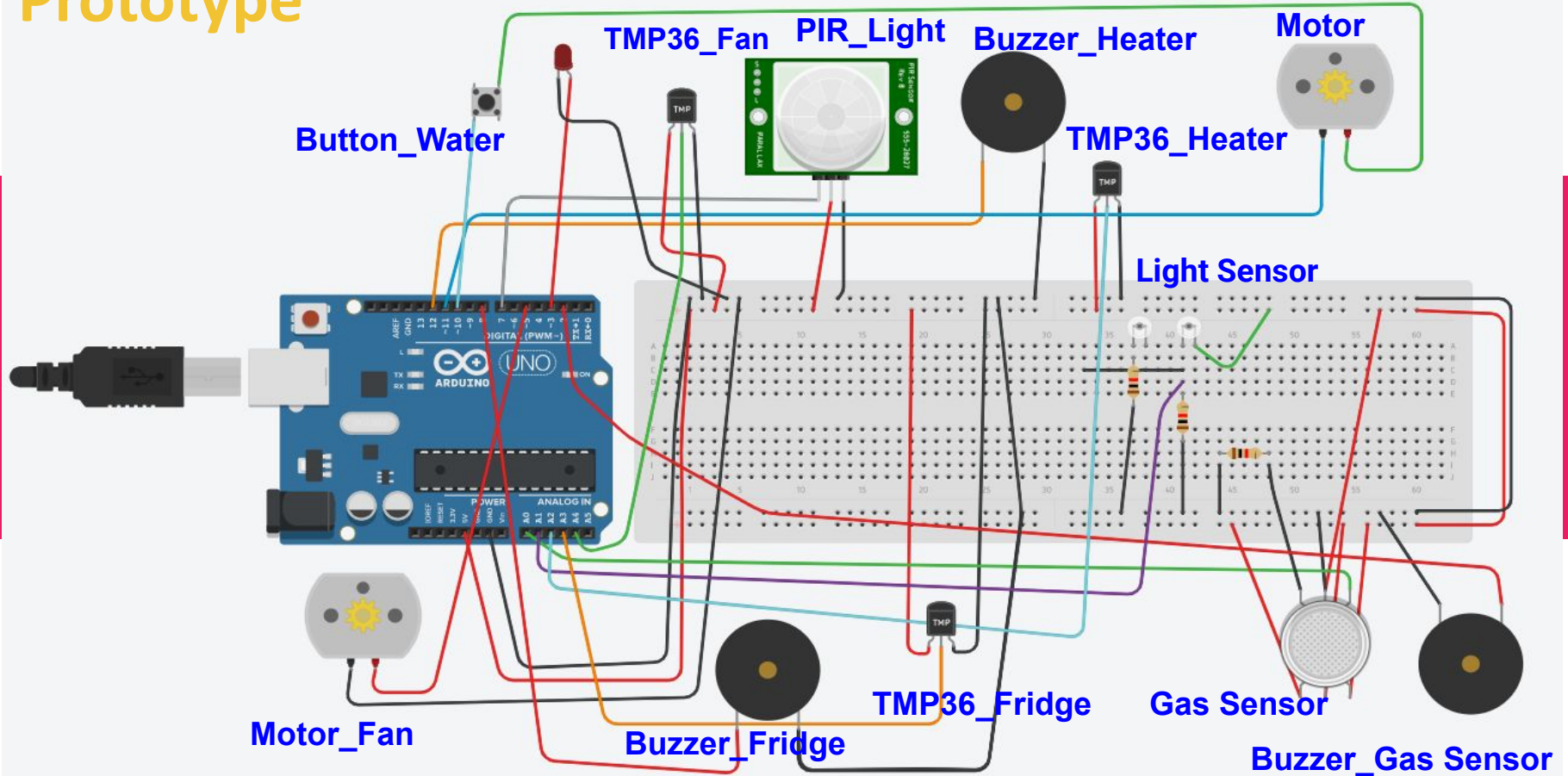
❖ **Light and Fan Control**
If user is entering the room, Both will turn on - PIR Sensor.
Fan's speed is controlled through temperature
Light's brightness is adjusted in par with sunlight. Power is saved

Block Diagram

Prototype

Button_Water

TMP36_Fan

PIR_Light

Buzzer_Heater

Motor

TMP36_Heater

Light Sensor

Motor_Fan

Buzzer_Fridge

TMP36_Fridge

Gas Sensor

Buzzer_Gas Sensor

Code

```cpp
// C++ code
//
int tmp36_heater=A2;
int buzzer_heater=12;
int motor=11;
int button=10;
int tmp36_fridge=A3;
int buzzer_fridge=8;
int pir=7;
int light_sensor=A1;
int fan_motor=5;
int tmp36_fan=A4;
int light_led=3;
int Gas_sensor = 0;
int buzzer_gas = 2;

void setup(
  Serial.begin(9600);
  pinMode(tmp36_heater,INPUT);
  pinMode(buzzer_heater,OUTPUT);
  pinMode(motor,OUTPUT);
  pinMode(button,INPUT);
  pinMode(tmp36_fridge,INPUT);
  pinMode(buzzer_fridge,OUTPUT);
  pinMode(pir,INPUT);
  pinMode(light_sensor,INPUT);
  pinMode(fan_motor,OUTPUT);
  pinMode(tmp36_fan,INPUT);
  pinMode(light_led,OUTPUT);
  pinMode(A0, INPUT);
  pinMode(buzzer_gas, OUTPUT);
}
```

```cpp
void heater()
{
   double sensor=-40+0.488155*(analogRead(tmp36_heater)-20);
  Serial.println("Enter the threshold temperature in Celsius\n");
  double threshold = Serial.read();
  if(sensor>threshold)
  {
    tone(buzzer_heater,253);
    Serial.println("Alert!!! Turn off heater");
  }
  else
    noTone(buzzer_heater);
}

void fridge()
{
  double sensor=-40+0.488155*(analogRead(tmp36_fridge)-20);
  Serial.println("Enter the threshold temperature in Celsius\n");
 double threshold = Serial.read();
  if(sensor>threshold)
    tone(buzzer_fridge,253);
  else
    noTone(buzzer_fridge);
}

void motor()
{
  int button_val=digitalRead(button);
   //water level goes down
   if(button_val==0)
   {
```

```cpp
void motor()
{
  int button_val=digitalRead(button);
    //water level goes down
    if(button_val==0)
    {
        for(int i=0;i<255;i++)
        {
            analogWrite(motor,i);
        }
    }
    //water level is full
    else
    {
        analogWrite(motor,0);
    }
}

void light()
{
  int pir_sensor=digitalRead(pir);
int light_val=analogRead(light_sensor);
if(pir_sensor==1)
{
    analogWrite(light_led,light_val);
    delay(10);
}
else
{
    analogWrite(light_led,0);
}
}
```

```cpp
void fan()
{
  int pir_sensor=digitalRead(pir);
double sensor=-40 + 0.488155 * (analogRead(tmp36_fan) - 20); //read input
 double rpm=sensor*2.4; //scaling
  Serial.print("The measured temp. value is ");
  Serial.println(sensor);
  if(sensor>20.00 && pir_sensor==1)
  {
    analogWrite(fan_motor,rpm);

  }
  else
  {
    analogWrite(fan_motor,0);
    Serial.println("The motor is now off");
  }
}

void gas()
{
Gas_sensor = analogRead(A0);
  if (Gas_sensor == 250)
  {
    tone(buzzer_gas, 523, 1000); // play tone 60 (C5 = 523 Hz)
  }
  delay(10); // Delay a little bit to improve simulation performance
}
```

```cpp
void loop()
{
 Serial.println("Enter your choice");
Serial.println("1.Heater Control, 2.Fridge Control, 3.Water Tank Motor Control, 4.Light and Fan Control, 5.Cylinder Control\n");
int choice=Serial.read();
switch(choice)
{
case 1:
heater();
break;

case 2:
fridge();
break;

case 3:
motor();
break;

case 4:
light();
fan();
break;

case 5:
gas();
break;

default:
Serial.println("Error\n");
}
}
```