TEAM ID : PNT2022TMID12783

PROJECT TITLE : Industry-Specific Intelligent Fire Management System

PROGRAM

#include "DHTesp.h"

#include <cstdlib> #include <time.h>

const int DHT\_PIN = 15;

bool is\_exhaust\_fan\_on = false; bool is\_sprinkler\_on = false;

float temperature = 0;

int gas\_ppm = 0;

int fire = 0; int flow = 0;

String fire\_status = "";

String accident\_status = "";

String sprinkler\_status = ""; DHTesp dhtSensor;

void setup() {

Serial.begin(99900);

dhtSensor.setup(DHT\_PIN, DHTesp::DHT22);

}

void loop() {

TempAndHumidity data = dhtSensor.getTempAndHumidity();

srand(time(0));

temperature = data.temperature; gas\_ppm = rand()%1000; int firereading = rand()%1024; fire = map(firereading,0,1024,0,1024); int firerange = map(firereading,0,1024,0,3); int flow = ((rand()%100)>50?1:0);

switch (firerange) { case 2: fire\_status = "Close Fire"; break; case 1: fire\_status = "Distant Fire"; break; case 0: fire\_status = "No Fire"; break;

}

if(gas\_ppm > 100){ is\_exhaust\_fan\_on = true;

}

else{ is\_exhaust\_fan\_on = false;

}

if(temperature < 40 && firerange ==2){ accident\_status = "need auditing"; is\_sprinkler\_on = false;

}

else if(temperature < 40 && firerange ==0){ accident\_status = "nothing found"; is\_sprinkler\_on = false;

}

else if(temperature > 50 && firerange == 1){ is\_sprinkler\_on = true; accident\_status = "moderate";

}

else if(temperature > 55 && firerange == 2){ is\_sprinkler\_on = true; accident\_status = "severe";

}else{ is\_sprinkler\_on = false; accident\_status = "nil"; }

if(is\_sprinkler\_on){ if(flow){

sprinkler\_status = "working";

}

else{

sprinkler\_status = "not working";

}

}

else if(is\_sprinkler\_on == false){ sprinkler\_status = "now it shouldn't";

}

else{

sprinkler\_status = "something's wrong";

}

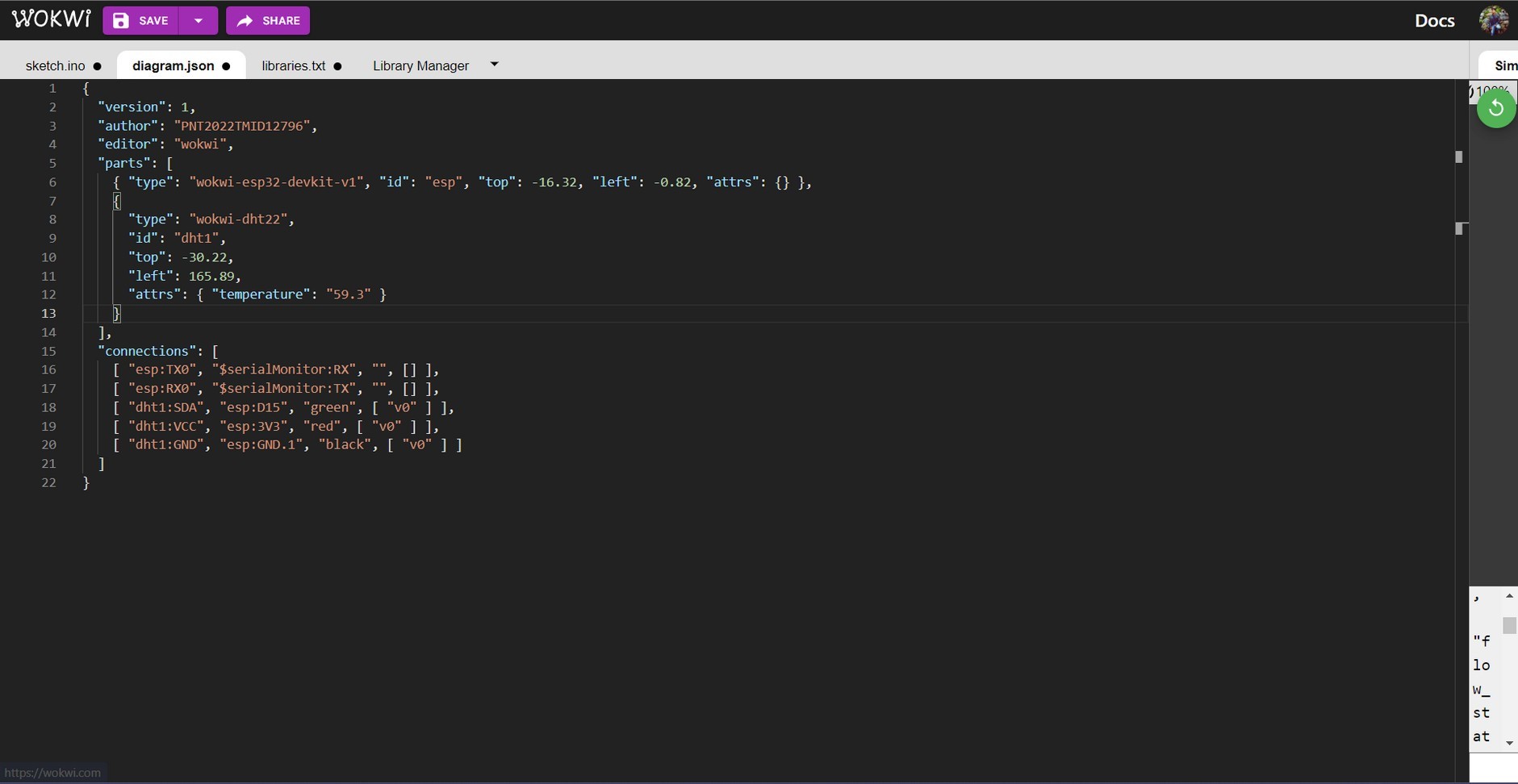
String out = "{\n\t\"senor\_values\":{"; out+="\n\t\t\"gas\_ppm\":"+String(gas\_ppm)+","; out+="\n\t\t\"temperature\":"+String(temperature,2)+","; out+="\n\t\t\"fire\":"+String(fire)+","; out+="\n\t\t\"flow\":"+String(flow)+",\n\t}"; out+="\n\t\"output\":{"; out+="\n\t\ t\"is\_exhaust\_fan\_on\":"+String((is\_exhaust\_fan\_on)?"true":"false")+","; out+="\n\t\t\"is\_sprinkler\_on\":"+String((is\_sprinkler\_on)?"true":"false")+","; out+="\n\t}";

out+="\n\t\"messages\":{"; out+="\n\t\t\"fire\_status\":"+fire\_status+","; out+="\n\t\t\"flow\_status\":"+sprinkler\_status+","; out+="\n\t\t\"accident\_status\":"+accident\_status+","; out+="\n\t}"; out+="\n}"; Serial.println(out);

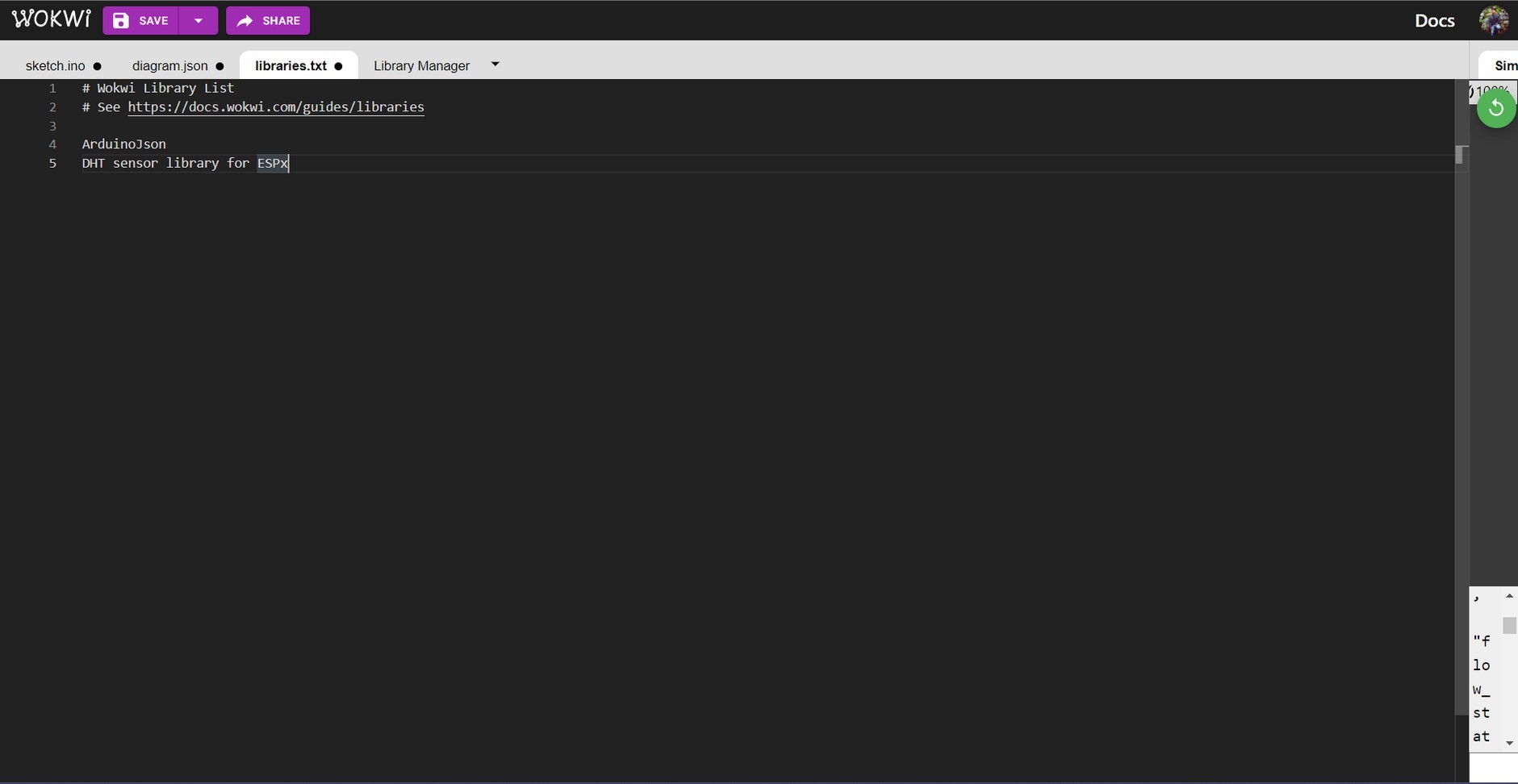
delay(1000);

}

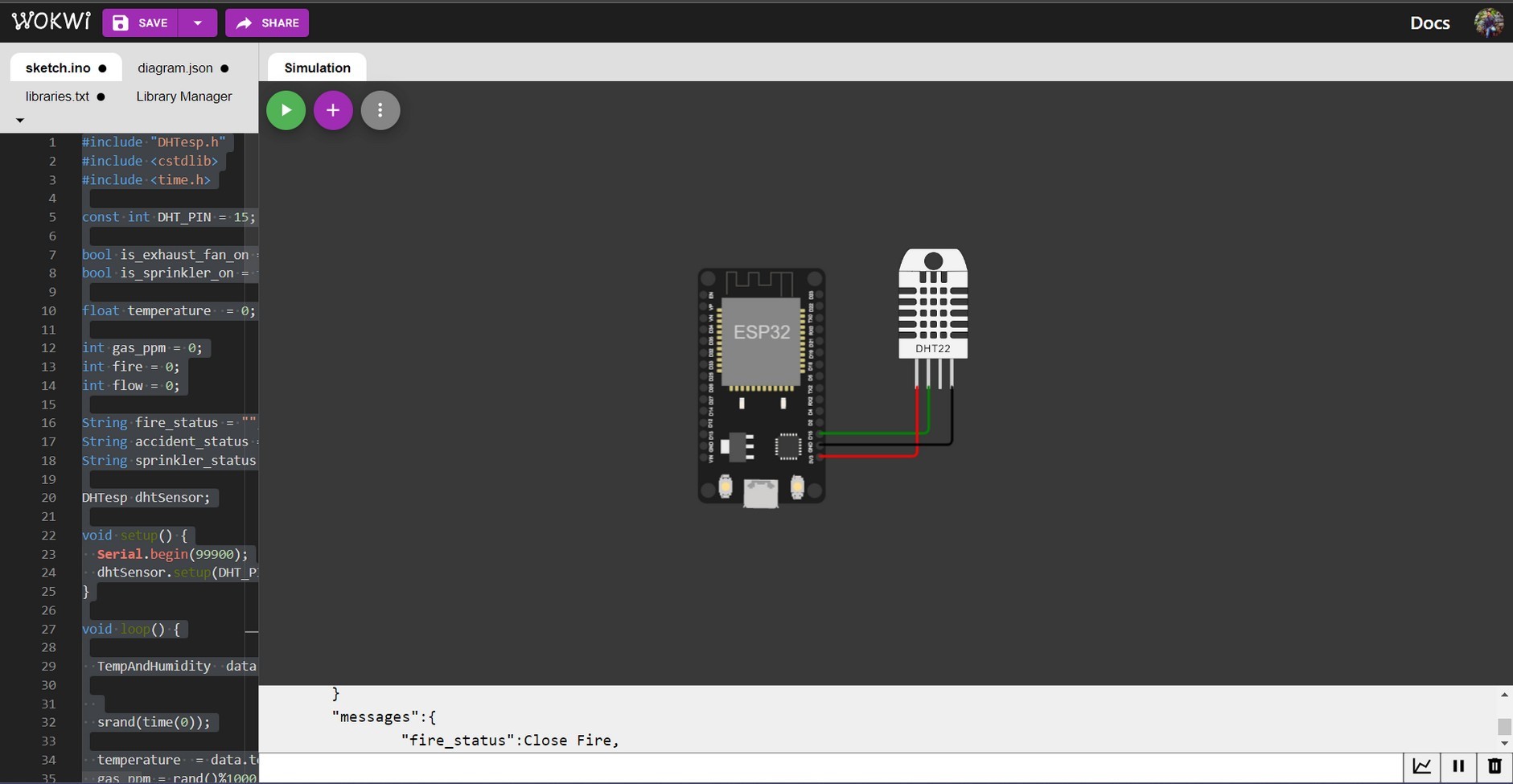
DIAGRAM.JSON:



LIBRARIES TEXT:



CIRCUIT:



OUTPUT:

