

SCORPION PROJECT WORKSHEET (PART 1)

Worksheet by Sebastian Yepez, June 2023

Hi, guys! I've loved working with all four of you for the past two weeks. But, now it's time to kick everything into final gear. Please answer all of the following questions thoughtfully and thoroughly.

Every team member must put their name on the document and also try to put as much input into this document as possible. Talk about the questions and answers and DON'T ask me for help unless something isn't clear! I really want you guys to get into the habit of working with your teammates, finding answers (even on the Internet), and thinking critically.

Names:

1. Eduardo
2. Amanda
3. Roma
4. Dylan

DAY 1:

Project Knowledge:

1. Ultimately, what is the goal of the project? What are we trying to build?

Answer : The ultimate goal of this project is to build a Multi-Purpose Smart Home.

2. Why are there 15 total "projects?"

Answer: There are 15 total projects in order to allow for a more organized schedule/project. It also helps to split up the days into specific tasks, which will help us complete the project on time.

3. When are we presenting the final project?

Answer: We are going to present the final project on June 23, 2023.

4. Where can you go to try finding more information on the subtask if you get stuck?

Answer: If we need to find more information on the subtask, we can refer to the wiki page for answers.

5. Why are we not using the included Arduino Plus board and shield that come with the kit?

Answer: We are not going to use the included Arduino Plus board and shield that comes with the kit because we have been using the Firebeetle ESP32-E board for the past week.

6. What will be one of the most challenging parts of the project? Think about the original board of the kit compared to the one we're using.

Answer: The programming and coding as well as getting the programming to work will be one of the most challenging parts of the project. Additionally, the switchover from the original board that comes with the kit to the ESP32, may cause difficulties with adapting the pin wiring from that to the ESP32.

7. What do S, V, and G stand for in the components of the Smart Home Kit? What do they correspond with on the Gravity IO Shield?

Answer: In the components of the Smart Home Kit, "S" stands for signal. "V" stands for positive, and G stands for ground.

8. What are Port PINs and the Arduino IDE equivalents? What is the Port PIN equivalent of D11? What is the Arduino equivalent of Port PIN 26?

Answer: Port PINs and Arduino IDE are two ways of describing the same thing. The Port PIN equivalent of D11 is "16". The Arduino equivalent of Port Pin 26 is "D3".

9. What will we be trying to accomplish today (Day 1)?

Answer: On the first day, we will try to fully understand/complete "Subtasks 1-9".

10. Give your individual responses below about what makes you most excited about this project:

1. Answer: Roma - Getting to compete with other teams/groups.
2. Answer: Dylan - Building the project.
3. Answer: Amanda - Working on the programming and getting it to work successfully.
4. Answer: Eduardo - Completing the project with the group.

Arduino Basic Knowledge:

1. What does the setup() function do?

Answer: The “setup() function” is called when a sketch starts and it is used to ultimately set up your Arduino for the rest of the program. It initializes things such as variables and pin modes.

2. How many times will loop() run?

Answer: A “loop()” is a loop that will continue to run until a condition is met. Therefore, it could run once or up to an infinite amount of times based on whether or not it is able to meet a condition.

3. How long will delay(5000) delay the program for?

Answer: A “delay(5000)” will delay the program for 5000 milliseconds, or 5 seconds.

4. What programming language is Arduino based on?

Answer: Arduino is based on C++ language.

5. What is a library?

Answer: A library is made up of classes and functions that are written in the core language of C++.

6. Why do you think we have to select a board and port before running the program?

Answer: The reason why we have to select a board before running the program is because the board selection tells Arduino how to actually compile the sketch as well as the protocols to use when it is uploading. We also need to select a port before running the programming because it tells the computer which physical board you will be using to run the program.

7. What do you think the error: “Property 'upload.tool.' is undefined” would relate to? (Hint: relates to last question).

Answer The error “Property 'upload.tool.' is undefined” would relate to a board/port not being connected to the program. Without a board or port being selected, the program would have an error trying to upload the program because it needs these components.

Subtask 1 - LED Blink:

1. What is the purpose of this subtask?

Answer: The purpose of this subtask was to program the LED to perform a series of different tasks. It was also used to help us learn more on how to use and implement loops and delays into our programming.

2. What number(s) did you have to replace in this program? Why?

Answer: We had to replace the numbers in the delays so that it could perform different tasks. For example, we had to change the numbers in the delay so that the light would turn off for ten seconds after blinking three times in one second.

3. Please give a brief explanation of how you think this program works. You may talk informally here if it helps. Pretend as if you were speaking to a friend.

Answer: We set the code so that the LED light would either stay on for 5 seconds and then stay off for 1. We put programming into a loop, and inside the loop we addressed the pin so that the LED light knows when to go on and off. We also added delays so that it can stay on for 5 seconds, and then stay off for 1. The loop will allow for this to happen one after another.

4. Please copy and paste your code for Task 1 here:

Answer: /*

```
Keyestudio smart home Kit for Arduino
Project 1
Blink
http://www.keyestudio.com
*/
void setup() {
    // initialize digital pin 12 (D12) as an output.
    pinMode(26, OUTPUT);
}
// the loop function runs over and over again forever
void loop() {
    digitalWrite(26, HIGH);    // turn the LED on (HIGH is the voltage level)
    delay(5000);               // wait for a second
    digitalWrite(26, LOW);     // turn the LED off by making the voltage LOW
    delay(1000);               // wait for a second
}
```

```
}//
```

5. Please copy and paste your code for Task 2 here:

Answer: /*

```
Keyestudio smart home Kit for Arduino
Project 1
Blink
http://www.keyestudio.com
*/
void setup() {
    // initialize digital pin 12 (D12) as an output.
    pinMode(26, OUTPUT);
}
// the loop function runs over and over again forever
void loop() {
    digitalWrite(26, HIGH);    // turn the LED on (HIGH is the voltage level)
    delay(100);                // wait for a second
    digitalWrite(26, LOW);     // turn the LED off by making the voltage LOW
    delay(100);                // wait for a second
}//
```

6. Please copy and paste your code for Task 3 here:

```
/*
Keyestudio smart home Kit for Arduino
Project 1
Blink
http://www.keyestudio.com
*/
void setup() {
    // initialize digital pin 12 (D12) as an output.
    pinMode(26, OUTPUT);
}
// the loop function runs over and over again forever
int count = 0;
void loop() {
```

```

    digitalWrite(26, HIGH); // turn the LED on (HIGH is the voltage
level)
    delay(1000);           // wait for a second
    digitalWrite(26, LOW); // turn the LED off by making the
voltage LOW
    delay(1000);           // wait for a second
    count++;
    if (count == 6) {
        delay(4000);
        count = 0;
    }
} //

```

Answer

Subtask 2 - Breathing Light:

1. What is the purpose of this subtask? What makes it different from LED Blink?

Answer: The purpose of this task was to make a program that allowed for the yellow LED light to stay at its brightest for four seconds after fading.

2. Why wouldn't the provided code work for this program? What code did you have to use?

Answer: The provided code wouldn't work for this program because it is based on Arduino Uno and we are using ESP32, which doesn't support analogWrite().

3. Google "What is PWM lights" and explain, in your own words, what PWM is. How does the duty cycle of the LED relate to this?

Answer: PWM is a common method of dimming that works by pulsing the lights very quickly which causes the light to appear as if it is a "steady dimmed light". The duty cycle of the LED relates to this because when the LED was fading, it was actually showing different intensities of light at a quick speed, making it appear to fade. This relates to what PWM is.

4. Explain how the for loops inside the program relate to Pulse Width Modulation and duty cycle.

Answer: The for loops inside the program relate to Pulse Width Modulation and duty cycle in the way that the code that makes the PWM LED works, by its very nature, must be in a loop in order to allow for the LED to work, as it functions in a loop of different intensities.

5. Please copy and paste your code for Task 1 here:

Answer

```
int freq = 1000;
int ledChannel = 0;
int resolution = 8;
#define LED_BUILTIN 26

void setup() {

    ledcSetup(ledChannel, freq, resolution);
    ledcAttachPin(LED_BUILTIN, ledChannel);

}

void loop() {

    for (int dutyCycle = 0; dutyCycle <= 255; dutyCycle++){
        ledcWrite(ledChannel, dutyCycle);
        delay(7);
    }
    delay(4000);

    for (int dutyCycle = 255; dutyCycle >= 0; dutyCycle--){
        ledcWrite(ledChannel, dutyCycle);
        delay(7);
    }

}
```

Subtask 3 - Passive Buzzer:

1. What is the purpose of this subtask?

Answer: The purpose of this subtask was to create a sound that has 3 frequencies, with the third being very long and deep.

2. Referring to the wiki, how can we use the buzzer to control different sounds? After skimming through [this](#) article, why is there a harsh, buzzy tone in the first program?

Answer: According to the wiki, "we can control different sound by inputting square waves with distinct frequency." The reason that there is a harsh, buzzy tone in the first program is because it contains a very sudden jump which causes high frequency harmonics in the sound spectrum.

3. What does delay(num) and i<80 do in the program? Try different numbers before you answer this question.

Answer: The delay measures the amount of time the high and low frequencies occur. The $i < 80$ gives the frequency a range.

4. After running the second program, try your best to give an explanation of what you think is going on.

Answer: After running the second program, the third harsh, deep frequency was made in a way that due to the longer delay, due to the very nature of the code, frequently buzzing in and out the passive buzzer, it had allowed for the passive buzzer to emit a longer and deeper third tone.

5. Please copy and paste your code for Task 1 here:

Answer:

```
/*  
Keyestudio smart home Kit for Arduino  
Project 3.1  
Buzzer  
http://www.keyestudio.com  
*/
```



```

int tonepin = 26; // Set the Pin of the buzzer to the digital D3
void setup ()
{
    pinMode (tonepin, OUTPUT); // Set the digital IO pin mode to output
}
void loop ()
{
    unsigned char i, j, m;
    for (i = 0; i <80; i ++) // output a frequency sound
    {
        digitalWrite (tonepin, HIGH); // Sound
        delay (1); // Delay 1ms
        digitalWrite (tonepin, LOW); // No sound
        delay (1); // Delay 1ms
    }
    for (j = 0; j <100; j ++) // output sound of another frequency
    { digitalWrite (tonepin, HIGH); // Sound
      delay (2); // delay 2ms
      digitalWrite (tonepin, LOW); // No sound
      delay (2); // delay 2ms
    }
    for (m = 0; m <100; m ++)// output a third frequency sound}}
    { digitalWrite (tonepin, HIGH);
      delay (10); // Delay 1000ms
      digitalWrite (tonepin, LOW);
      delay (10); // delay 1000ms
    }
}
}

```

Subtask 4 - Controlling LED by Button Module+:

1. What is the purpose of this subtask? Why is it important to know how a physical input can give us a digital output?

Answer: The purpose of this subtask is to implement a physical input, the button, in order to enact a digital output, the LED in a certain way.

2. Explain in your own words how val and the if-else statement are working hand in hand to give an output to the LED.

Answer: The purpose of “val” is for the computer to check and interpret the conditions of the programming. It is working hand in hand with the if-else statement due to the fact that it is checking the state of the hardware, allowing for the computer to interpret whether or not it needs to turn off or on based on the physical input.

3. What was the most challenging part about Task 2?

Answer: The most challenging part of Task 2 was trying to figure out how to make the programming work.

4. Please copy and paste your code for Task 1 here:

Answer: / *

```
Keyestudio smart home Kit for Arduino
Project 4
Button
http://www.keyestudio.com
* /
int ledpin = 26; // Define the led light in D5
int inpin = 0; // Define the button in D4
int val; // Define variable val
void setup ()
{
  pinMode (ledpin, OUTPUT); // The LED light interface is defined as output
  pinMode (inpin, INPUT); // Define the button interface as input
}
void loop ()
{
  val = digitalRead (inpin); // Read the digital 4 level value and assign it
  to val
  if (val == HIGH ) // Whether the key is pressed, the light will be on when
  pressed
  {digitalWrite (ledpin, HIGH);}
  else
  {digitalWrite (ledpin, LOW);}
}

}
```

5. Please copy and paste your code for Task 2 here:

```
#define NTD0 -1
#define NTD1 294
#define NTD2 330
#define NTD3 350
#define NTD4 393
#define NTD5 441
#define NTD6 495
#define NTD7 556

#define NTDL1 147
#define NTDL2 165
#define NTDL3 175
#define NTDL4 196
#define NTDL5 221
#define NTDL6 248
#define NTDL7 278

#define NTDH1 589
#define NTDH2 661
#define NTDH3 700
#define NTDH4 786
#define NTDH5 882
#define NTDH6 990
#define NTDH7 112
// List all D-tuned frequencies
#define WHOLE 1
#define HALF 0.5
#define QUARTER 0.25
#define EIGHTH 0.25
#define SIXTEENTH 0.625
// List all beats
int tune [] = // List each frequency according to the notation
{
    NTD3, NTD3, NTD4, NTD5,
    NTD5, NTD4, NTD3, NTD2,
    NTD1, NTD1, NTD2, NTD3,
    NTD3, NTD2, NTD2,
    NTD3, NTD3, NTD4, NTD5,
```

```

    NTD5, NTD4, NTD3, NTD2,
    NTD1, NTD1, NTD2, NTD3,
    NTD2, NTD1, NTD1,
    NTD2, NTD2, NTD3, NTD1,
    NTD2, NTD3, NTD4, NTD3, NTD1,
    NTD2, NTD3, NTD4, NTD3, NTD2,
    NTD1, NTD2, NTDL5, NTD0,
    NTD3, NTD3, NTD4, NTD5,
    NTD5, NTD4, NTD3, NTD4, NTD2,
    NTD1, NTD1, NTD2, NTD3,
    NTD2, NTD1, NTD1
};

float durt [] = // List the beats according to the notation
{
    1,1,1,1,
    1,1,1,1,
    1,1,1,1,
    1 + 0.5,0.5,1 + 1,
    1,1,1,1,
    1,1,1,1,
    1,1,1,1,
    1 + 0.5,0.5,1 + 1,
    1,1,1,1,
    1,0.5,0.5,1,1,
    1,0.5,0.5,1,1,
    1,1,1,1,
    1,1,1,1,
    1,1,1,0.5,0.5,
    1,1,1,1,
    1 + 0.5,0.5,1 + 1,
};

int length;
int tonepin = 16; // Use interface 3
int ledpin = 26; // Define the led light in D3
int inpin = D5; // Define the button in D5
int val; // Define variable val

void setup ()
{
    //button

```

```

pinMode (ledpin, OUTPUT); // The LED light interface is defined as
output
pinMode (inpin, INPUT); // Define the button interface as input

//buzzer
pinMode (tonepin, OUTPUT);
length = sizeof (tune) / sizeof (tune [0]); // Calculate length
}
void odetoJoy ()
{
    for (int x = 0; x <length; x ++)
    {
        tone (tonepin, tune [x]);
        delay (350* durt [x]); // This is used to adjust the delay
according to the beat, 350 can be adjusted by yourself.
        noTone (tonepin);
    }
    delay (2000); // delay 2S
}

void loop ()
{
    val = digitalRead (inpin); // Read the digital 4 level value and
assign it to val
    if (val == LOW) // Whether the key is pressed, the light will be on
when pressed
    {digitalWrite (ledpin, HIGH);
    odetoJoy();
    }
    else
    {digitalWrite (ledpin, LOW);}
    val = digitalRead (inpin);
}

```

Subtask 5 - 1-Channel Relay Module

1. What is the purpose of this subtask?

Answer: The purpose of this subtask is to familiarize ourselves with the relay module and its functions.

2. What is a relay module? What is the main message I wanted to get across?

Answer: The relay module is used to relay the signal it receives from the board to the component that it is wired to. The main message you wanted to get across the relay module passes a current to another module.

3. Go to [this website](#) and research on Google about relay modules. Do your best to explain how they work. What does NO stand for? What about COM or NC? What is the purpose of the electromagnet?

Answer: Relay modules are switches that use electricity and are operated by an electromagnet. That electromagnet is controlled by a signal from a microcontroller. When it is activated, this magnet can open or close an electrical circuit. This allows for a current to be carried from one module to another.

4. Were you able to see your LED light up? Try your best to look from different angles.

Mine was very dim sometimes to the point where I could only see it from a certain perspective, and sometimes it was bright depending on the angle of the pin in the NO contact terminal. Fidget with it and see if you get a response.

Answer: Yes, we were able to see the LED light up.

```
/*
Keyestudio smart home Kit for Arduino
Project 5
Relay
http://www.keyestudio.com
*/
int Relay = 4; // Define the relay pin at D12
void setup ()
{
  pinMode (26, OUTPUT); // Set Pin13 as output
  digitalWrite (26, HIGH); // Set Pin13 High
  pinMode (Relay, OUTPUT); // Set Pin12 as output
```

```
}  
void loop ()  
{  
  digitalWrite (Relay, HIGH); // Turn off relay  
  delay (2000);  
  digitalWrite (Relay, LOW); // Turn on relay  
  delay (2000);  
}
```

Subtask 6 - Photocell Sensor

1. What is the purpose of this subtask?

Answer: The purpose of this subtask was to change a line in the programming so that the LED is automatically on until a certain light level is reached.

2. According to the wiki, what is a photocell sensor and how does it work? If you need to, use Google to find more information and do your best to describe the process.

Answer: A photocell sensor is a resistor that when light is shone onto it, it shoots out electrons which is the photoelectric effect. The resistance of that process is what is used to measure the light.

3. What is the difference between an analog and digital signal? Why does this sensor need to be attached to an analog port on the microcontroller (ESP32)?

Answer: Analog and digital signals are both types of signals that carry information. However, analog signals have continuous electrical signals, while digital signals have non-continuous electrical signals.

4. What real life implementations do you think this kind of sensor could be used for? List three examples and go into detail about each of them.

Answer: One real life implementation we think this sensor could be used for is in a smart home, this type of sensor could recognize when the lights are on/off and control them based on the preference of a homeowner.

5. What is a photodiode? Give a very brief description and state how it is implemented into the code.

Answer: Photodiodes are semiconductors that are utilized within the photocell sensor that utilizes the photoelectric effect, causing absorbed photons to produce a flow of electrons. This is implemented in the code, as the resistance produced within this process is used to measure light levels.

6. Please copy and paste your code for Task 1 here:

Answer

```
int LED = 16; // Set LED pin at D5
int val = 0; // Read the voltage value of the photodiode
void setup () {
    pinMode (LED, OUTPUT); // LED is output
    Serial.begin (115200); // The serial port baud rate is set to 9600
}
void loop () {
    val = analogRead (A1); // Read the voltage value of A1 Pin
    Serial.println (val); // Serial port to view the change of voltage
    value
    if (val >900)
    {
        // Less than 900, LED light is off
        digitalWrite (LED, LOW);
    }
    else
    {
        // Otherwise, the LED lights up
        digitalWrite (LED, HIGH);
    }
    delay (10); // Delay 10ms
}
//
```


Subtask 7 - Adjusting Motor Servo Angle

1. What is the purpose of this subtask? Hint: read the wiki for this one.

Answer: To experiment with using servo motors (a position control rotary actor).

What is the max rotation angle the servo motor can do? Try it out for yourself and then look on the wiki for confirmation.

Answer: The max rotation angle a servo can accomplish is a 180 degree turn.

2. Why can only digital pins 9 and 10 be used for this program? Hint: Look at the wiki.

Answer: Because the *servo function* exists, only digital pin 9 and 10 can be used.

3. What was the compilation error you ran into when trying the program for the first time?

How did you fix it?

Answer: The motor wouldn't run and it would often stop. We got rid of the for loop and set a (1000) delay.

4. Please copy and paste your code for Task 1 here:

```
5.  #include <ESP32Servo.h>
6.
7.  Servo myservo;  // create servo object to control a servo
8.  // 16 servo objects can be created on the ESP32
9.
10. int pos = 0;    // variable to store the servo position
11. // Recommended PWM GPIO pins on the ESP32 include
    2,4,12-19,21-23,25-27,32-33
12. int servoPin = 17;
13.
14. void setup() {
15.  myservo.setPeriodHertz(50);    // standard 50 hz servo
16.  myservo.attach(servoPin, 500, 2400); // attaches the servo on pin
    18 to the servo object
17.  // using default min/max of 500us and 2400us
18.  // different servos may require different min/max settings
```

```

19. // for an accurate 0 to 180 sweep
20. }
21.
22. void loop() {
23.
24.   for (pos = 0; pos <= 180; pos += 1) { // goes from 0 degrees to
      180 degrees
25.     // in steps of 1 degree
26.     myservo.write(pos);    // tell servo to go to position in
      variable 'pos'
27.     delay(5);              // waits 15ms for the servo to reach the
      position
28.   }
29.   delay(1000);
30.   for (pos = 180; pos >= 1; pos -= 1) { // goes from 180 degrees to
      0 degrees
31.     myservo.write(pos);    // tell servo to go to position in
      variable 'pos'
32.     delay(5);              // waits 15ms for the servo to reach the
      position
33.   }
34. }
35.

```

Subtask 8 - Fan Module

1. What is the purpose of this subtask?

Answer: The purpose of this subtask thus far is to use the program to make the fan spin.

2. Why can this fan control its direction and speed?

Answer: The fan can control its direction and speed because it has a motor control trip that allows it to do so.

3. Looking and testing the code provided, why are there two signal wires connected to D6 and D7? What do they do?

Answer: The reason why there are two signal wires connected to D6 and D7 is to control the direction of the fan.

4. After having done the task, please copy and paste your code here:

```
Answer void setup () {  
  
    pinMode (13, OUTPUT); //define D7 pin as output  
    pinMode (14, OUTPUT); //define D6 pin as output  
}  
void loop () {  
    digitalWrite (13, HIGH);  
    digitalWrite (14, LOW); // Reverse rotation of the motor  
    delay (5000); // delay 3S  
    digitalWrite (13, LOW);  
    digitalWrite (14, LOW); // The motor stops rotating  
    delay (5000); //delay 1S  
    digitalWrite (13, LOW);  
    digitalWrite (14, HIGH); // The motor rotates in the forward direction  
    delay (10000); // delay 3S  
}
```

Subtask 9 - Steam Sensor

1. What is the purpose of this subtask?

Answer: The purpose of this subtask was to connect the sensor properly and change the programming so that the sensor is able to properly detect moisture levels .

2. What are some of the implications this component can have on your final project and the real world?

Answer: One real implication that this component can have on our final project is using the sensor to detect moisture levels of plants.

3. Do you think the steam sensor would be able to sense boiling water? Why or why not?
Should we try it?

Answer: We think that the steam sensor would be able to sense boiling water due to the steam that comes off from boiling water. Sure, we could try it.

Subtask 10 - PIR Motion Detector

1. What is the purpose of this subtask?

Answer: The purpose of this subtask is to get a buzz to play from the passive buzzer whenever the motion sensor detects motion.

2. What are some of the implications this component can have on your final project and the real world?

Answer: One main implication this component can have on our final project is using this to represent a fire in some situation. For example, if the motion sensor detects fire, it will set off the fan to help better the situation, but also set off the buzzer which can kind of act as a fire alarm.

3. How does the PIR Motion Sensor detect if something is in front of it?

Answer: The PIR motion sensor is able to detect if something is in front of it by using infrared signals that come off from a moving person or animal.

4. Please copy and paste your code for Task 1 here:

Answer

```
void setup () {  
  
  Serial.begin (115200); // open serial port, and set baud rate at 9600bps  
  pinMode (25, INPUT); // Define PIR as input in D2  
  pinMode (12, OUTPUT);  
  pinMode (26, OUTPUT); // Define LED as output in D13  
  pinMode (13, OUTPUT); // Define D7 as output  
  pinMode (14, OUTPUT); // Define D6 as output  
}  
  
void loop () {  
  Serial.println (digitalRead (D2));  
  delay (500); // Delay 500ms  
  if (digitalRead (D2) == 1) // If someone is detected walking  
  {  
    digitalWrite (26, HIGH); // LED light is on  
  }  
}
```

```

digitalWrite (13, HIGH);
digitalWrite (14, LOW);
// Fan rotates
int i;
for (i = 0; i <80; i ++) // output a frequency sound
{
    digitalWrite (12, HIGH); // Sound
    delay (1); // Delay 1ms
    digitalWrite (12, LOW); // No sound
    delay (1); // Delay 1ms
}
} else // If no person is detected walking
{
digitalWrite (26, LOW); // LED light is not on
digitalWrite (13, LOW);
digitalWrite (14, LOW);} }

```

Subtask 11 - Gas Sensor

1. What is the purpose of this subtask?

Answer: The purpose of this subtask was to make a functioning gas sensor that also works as a smoke sensor.

2. What are some of the implications this component can have on your final project and the real world?

Answer: Similar to the last task, one implication we can have on this final project and in the real world is using this to detect a situation in fire.

3. What does PPM mean in this context?

Answer: PPM is the measurement of concentration in certain material particles.

After completing Task 1, share your observations.

Answer: We noticed that it took a while for it to start buzzing. However after it was heated, the buzzer went off when the lighter was next to the gas sensor

Subtask 12 - LCD Display

1. What is the purpose of this subtask?

Answer: To display a printed message on the LCD Screen.

2. What are some of the implications this component can have on your final project and the real world?

Answer: One implication this component can have in our final project is by somehow linking the LCD with the motion detector so that if motion is sensed, the LCD screen could display a warning message.

3. What does I2C mean? Use the Internet.

Answer: I2C is an interface used to communicate from a controller to any other device known as a "slave device".

4. Please copy and paste your code from Task 1 here:

Answer: `#include <Wire.h>`

```
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd (0x27,16,2); // set the LCD address to 0x27 for a16
chars and 2 line display
void setup ()
{
  lcd.init (); // initialize the lcd
  lcd.init (); // initialize the lcd
  // Print a message to the LCD.
  lcd.backlight ();
  lcd.setCursor (0,0);
  lcd.print ("Hey!"); // LED print hello, world!
  lcd.setCursor (0,1);
  lcd.print ("Vegas Stem Lab!"); // LED print keystudio!
}
void loop ()
{
}
```

Subtask 13 - Soil Humidity Sensor

1. What is the purpose of this subtask?

Answer: The purpose of this subtask was to measure the water moisture level of the soil.

2. What are some of the implications this component can have on your final project and the real world? Please answer thoroughly.

Answer: Some implications that this component can have on our final project is it can detect weather or not a plant needs water, to ensure the best quality conditions for the plant to thrive.

3. What happened when you put water into the soil?

Answer: The program detected whether or not the soil was in water as well as its moisture level.