SPRINT-1

**Project Name: SMART FARMER: IoT Enabled Smart Farming Application**

**Team ID:PNT2022TMID42977**

PROGRAM:

/\*

Plant Watering System

The circuit:

* Water pump

Power supply: 4.5~12V DC

Interface: Brown +; Blue - Temperature/moisture sensor Power supply: 3.3-5v

* Moisture sensor

Power supply: 3.3-5v

\*/

#include "DHT. h"

#defineDHTPIN 2 // what digital pin we're connected to

#defineDHTTYPE DHT22 // DHT 22 (AM2302), AM2321

DHT dh t (DHTPIN, DHTTYPE);

const int SOIL\_MOISTURE\_SENSOR\_PIN = A0; const int WATER\_PUMP\_PIN = 4;

const int dry = 520; const int wet = 270; const int moisture Levels = (dry - wet) /3;

// TODO: Should we have a counter so if it waters for X times, then take a break?

// OPTIMIZE: how dry to start watering and for how long.

const int soil Moisture art Watering = 400; const int soil Moisture Stop Watering = 300;

// 60 seconds const long water Duration = 1000L \* 60L;

// 60 seconds const long sensor Read Intervals = 1000L \* 60L;

// 2hr const long water Intervals = 1000L \* 60L \* 60L \* 2; long last Water Time = -water Intervals - 1; boo lean is Watering = false;

void setup ()

{Serial. begin(9600);

pin Mode (WATER\_PUMP\_PIN, OUTPUT);

water Pump Off (); dh t. Begin ();

}

void loop ()

{main Loop

();

}

void main Loop () {

float temperature = get Temperature ();

float humidity = get Humidity ();

long soil Moisture = analog Read (SOIL\_MOISTURE\_SENSOR\_PIN);

Serial. Print ln ("Soil Moisture: " + readable Soil Moisture (soil Moisture) + ", " + soil Moisture);

Serial. Print ln ("Temperature: " + String(temperature) + " \*F");

Serial. Print ln ("Humidity:

" + String(humidity) + " %");

if (mil l is () – last Water Time > water Intervals)

{water Plants (soil Moisture);

last Water Time = mil l is ();

}

Delay (sensor Read Intervals);

}

void water Plants (int soil Moisture) {

// Should this take a moving avg of the soil Moisture?

// Can get outliers on the right after watering. if (soil Moisture > soil Moisture art Watering)

{is Watering = true;

} else if (soil Moisture < soil Moisture Stop Watering)

{is Watering = false;

}

Serial. Print ln (is Watering? "Starting to water": "Skipping water");

if (is Watering) {water Pump On ();

delay (water Duration);

water Pump Off ();

Serial. Print ln ("Done watering");

}

}

String readable Soil Moisture (int soil Moisture)

{if (soil Moisture <= wet) {

return "Water";

} else if (soil Moisture > wet && soil Moisture < (wet + moisture Levels))

{return "Very Wet";

} else if (soil Moisture > (wet + moisture Levels) && soil Moisture < (dry moisture Levels)) {return "Wet";

} else if (soil Moisture < dry && soil Moisture > (dry – moisture Levels)) {return "Dry";

} else

{return

"Air";

}

}

float get Temperature () {

// Read temperature as Fahrenheit (is Fahrenheit = true) float temperature = dh t. Read Temperature(true);

if (is nan(temperature)) {

Serial. Print ln ("Failed to read from DHT sensor!");

} return temperature;

}

float get Humidity () {float humidity = dh t. read Humidity ();

if (is nan(humidity))

{

Serial. Print ln ("Failed to read from DHT sensor!");

} return humidity;}

void water Pump On ()

{Serial. print ln("Water pump on");

digital Write (WATER\_PUMP\_PIN, LOW);

}

void water Pump Off ()

{Serial print ln ("Water pump off");

digital Write (WATER\_PUMP\_PIN, HIGH);

}