IoT Arduino Workflow Document

Introduction to Arduino

- Arduino is an open-source hardware, software and content platform with a global community.
- SoC means software on chip in which software is installed on a single chip

LCD

- Connect 10 pin FRC cable between CN3 to CN6

```
#include <LiquidCrystal.h>

// Define LCD connections
const int rs = 6, en = 7, d4 = 5, d5 = 4, d6 = 3, d7 = 2;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

void setup() {
  // Initialize LCD
  lcd.begin(16, 2);
}

void loop() {
  // Print moisture percentage on LCD
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Hello All!");
  lcd.setCursor(0, 10);
  // Delay before taking next measurement
  delay(1000);
}
```

7 Segment Display

- Connect 10 pin FRC cable between CN2-CN8

```
#include "SevSeg.h"

SevSeg sevseg; // Instantiate a seven-segment controller object

void setup() {
  byte numDigits = 1;
  byte digitPins[] = {2};
  byte segmentPins[] = {3, 4, 5, 6, 7, 8, 9, 10};
  bool resistorsOnSegments = false; // 'false' means resistors are on digit
pins
  byte hardwareConfig = COMMON_ANODE; // See README.md for options
  bool updateWithDelays = false; // Default 'false' is Recommended
```

```
  bool leadingZeros = false; // Use 'true' if you'd like to keep the leading
zeros
  bool disableDecPoint = true; // Use 'true' if your decimal point doesn't
exist or isn't connected

  sevseg.begin(hardwareConfig, numDigits, digitPins, segmentPins,
resistorsOnSegments,
               updateWithDelays, leadingZeros, disableDecPoint);
  sevseg.setBrightness(90);
}

void loop() {
  // Read the input from a potentiometer
  int sensorValue = analogRead(A0);

  // Map the sensor value to a range suitable for a 1-digit display (0-9)
  int displayValue = map(sensorValue, 0, 1023, 0, 9);

  // Display the mapped value on the 7-segment display
  sevseg.setNumber(displayValue);
  sevseg.refreshDisplay();

  // Add a delay for better readability
  delay(200);
}
```

Keypad

- Connect 10 pin FRC cable between CN9-CN5.

```
  #include <Keypad.h>


  const byte ROWS = 4;
  const byte COLS = 4;

  char hexaKeys[ROWS][COLS] = {
    {'1', '2', '3', 'A'},
    {'4', '5', '6', 'B'},
    {'7', '8', '9', 'C'},
    {'*', '0', '#', 'D'}
  };

  byte rowPins[ROWS] = {9, 8, 7, 6};
  byte colPins[COLS] = {5, 4, 3, 2};

  Keypad customKeypad = Keypad(makeKeymap(hexaKeys), rowPins, colPins, ROWS,
COLS);
```

```
  void setup(){
    Serial.begin(9600);
  }

  void loop(){
    char customKey = customKeypad.getKey();

    if (customKey){
      Serial.println(customKey);
    }
  }
```

Soil Moisture Sensor

- Connect 4 pin relimate cable between RM23 – RM13.

```
// Define soil moisture sensor pin
const int moisturePin = A0;

void setup() {

  // Set soil moisture sensor pin as input
  pinMode(moisturePin, INPUT);
}

void loop() {
  // Read soil moisture value
  int moistureValue = analogRead(moisturePin);

  // Map the moisture value to a percentage (0-100%)
  int moisturePercentage = map(moistureValue, 0, 1023, 0, 100);


  printf("Soil Moisture:");
  printf(moisturePercentage);
  printf("%");

  // Delay before taking next measurement
  delay(1000);
}
```

Gas Sensor

- Connect 4 pin relimate cable between RM22-RM12

```
// Define the analog pin connected to the gas sensor
const int gasSensorPin = A0;
```

```
void setup() {
  Serial.begin(9600);
}

void loop() {
  // Read the analog value from the gas sensor
  int sensorValue = analogRead(gasSensorPin);

  // Convert the analog value to a voltage (0-5V)
  float voltage = sensorValue * (5.0 / 1023.0);

  // Print the raw sensor value and voltage to the serial monitor
  Serial.print("Raw Sensor Value: ");
  Serial.print(sensorValue);
  Serial.print("\tVoltage: ");
  Serial.print(voltage);
  Serial.println("V");

  // Add your logic to interpret the gas concentration based on the sensor
values
  // You may need to calibrate the sensor and refer to its datasheet for
specific details

  delay(1000); // Adjust the delay based on your application needs
}
```

Ultrasonic Sensor

- Connect 4 pin relimate cable between RM21 to RM4.

```
const int pingPin = 7;

void setup() {
  // initialize serial communication:
  Serial.begin(9600);
}

void loop() {
  // establish variables for duration of the ping, and the distance result
  // in inches and centimeters:
  long duration, inches, cm;

  // The PING))) is triggered by a HIGH pulse of 2 or more microseconds.
  // Give a short LOW pulse beforehand to ensure a clean HIGH pulse:
  pinMode(pingPin, OUTPUT);
  digitalWrite(pingPin, LOW);
  delayMicroseconds(2);
```

```
  digitalWrite(pingPin, HIGH);
  delayMicroseconds(5);
  digitalWrite(pingPin, LOW);
```

Real Time Clock

- Connect 4 pin relimate cable between RM1-RM10.

```
#include <RTClib.h>

RTC_DS3231 rtc;

char daysOfTheWeek[7][12] = {
  "Sunday",
  "Monday",
  "Tuesday",
  "Wednesday",
  "Thursday",
  "Friday",
  "Saturday"
};

void setup () {
  Serial.begin(9600);

  // SETUP RTC MODULE
  if (! rtc.begin()) {
    Serial.println("Couldn't find RTC");
    Serial.flush();
    while (1);
  }

  // automatically sets the RTC to the date & time on PC this sketch was
compiled
  rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));

  // manually sets the RTC with an explicit date & time, for example to set
  // January 21, 2021 at 3am you would call:
  // rtc.adjust(DateTime(2021, 1, 21, 3, 0, 0));
}

void loop () {
  DateTime now = rtc.now();
  Serial.print("Date & Time: ");
  Serial.print(now.year(), DEC);
  Serial.print('/');
  Serial.print(now.month(), DEC);
  Serial.print('/');
  Serial.print(now.day(), DEC);
```

```
  Serial.print(" (");
  Serial.print(daysOfTheWeek[now.dayOfTheWeek()]);
  Serial.print(") ");
  Serial.print(now.hour(), DEC);
  Serial.print(':');
  Serial.print(now.minute(), DEC);
  Serial.print(':');
  Serial.println(now.second(), DEC);

  delay(1000); // delay 1 seconds
}
```

Buzzer

- Connect 4 pin relimate cable between RM25 – RM26, RM17 – RM9

```
const int buzzer = 9; //buzzer to arduino pin 9

void setup(){
  pinMode(buzzer, OUTPUT); // Set buzzer - pin 9 as an output
}

void loop(){
  tone(buzzer, 1000); // Send 1KHz sound signal...
  delay(100);        // ...for 1 sec
  noTone(buzzer);     // Stop sound...
  delay(100);        // ...for 1sec
}
```

Light Sensor

- Connect 4 pin relimate cable between RM23 – RM13

```
void setup() {
  // Setup serial communication at baudrate 9600 for reading the light sensor
  Serial.begin(9600);
}

void loop() {
  // reads the input on analog pin A0
  int lightValue = analogRead(A0);

  // Print out the values to read in the Serial Monitor
  Serial.print("Analog reading (0-1023): ");
  Serial.print(lightValue);
```

```
  // Use the value to determine how dark it is
  // (Try tweaking these to make it more accurate)
  if (lightValue < 10) {
    Serial.println(" - Dark");
  } else if (lightValue < 200) {
    Serial.println(" - Dim");
  } else if (lightValue < 500) {
    Serial.println(" - Light");
  } else if (lightValue < 800) {
    Serial.println(" - Bright");
  } else {
    Serial.println(" - Very bright");
  }

  delay(500);
}
```

LED

- Connect 10 pin FRC cable between CN9-CN4

```
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH);  // turn the LED on (HIGH is the voltage
level)
  delay(1000);                      // wait for a second
  digitalWrite(LED_BUILTIN, LOW);   // turn the LED off by making the voltage
LOW
  delay(1000);                      // wait for a second
}
```