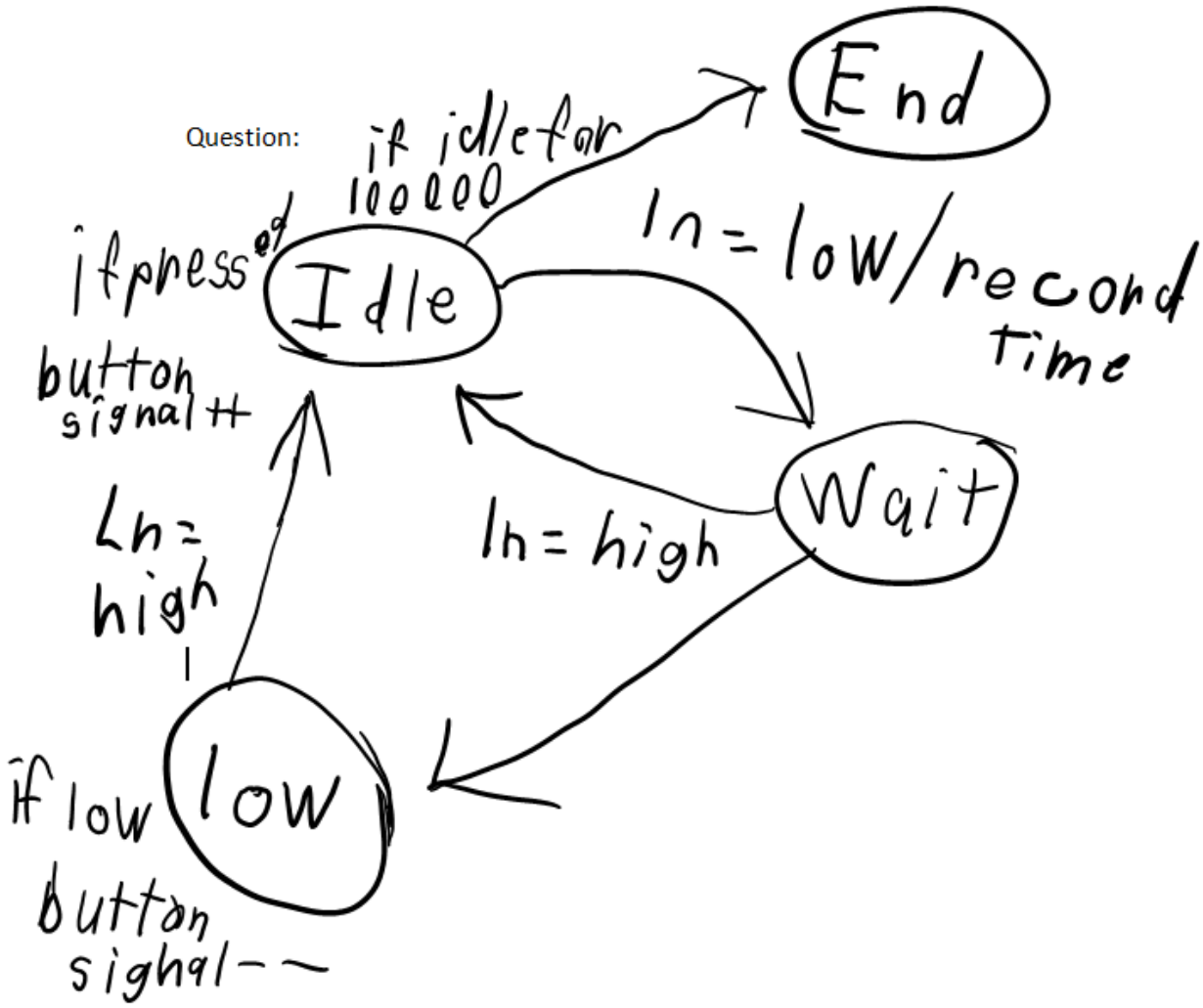
TERRAN BLAKE // WEDNESDAY 7:30AM // ECE 241

BLUE = PART 1 RED = BOTH PARTS GREEN = PART 2




```
enum ProcStates { Idle, Wait2, Low2};

#include<LiquidCrystal.h>                          //These two lines setup the drivers for the display
LiquidCrystal LcdDriver(12, 11, 5, 6, 7, 8);

int interval = 5;
int ProcStates;
int buttonPin = 4;
int buttonPressCount = -1;
int tracker = 0;
int debounceInterval = 5;                          //Lets the button settle for 5ms

unsigned long currenttime = 0;
unsigned long timer = 0;                    //integer for button pin
unsigned long timeOfLastButtonEvent = 0;                    //store the last time the button state changed

boolean Input = LOW;                        //Stores the value of the current state
boolean Wait = LOW;                          //Stores the last state so that it loops properly
boolean Low = LOW;
```

```
void setup() {
  pinMode(buttonPin, INPUT);                                    //Sets pin 4 as an Input
  Serial.begin(9600);                                          //Begins Lcd on pin 16 and 2
  LcdDriver.begin(16, 2);
  LcdDriver.clear();
  LcdDriver.setCursor(0,0);
  ProcStates = Idle;
  pinMode(10,OUTPUT);
}

int NextState(int ProcStates) {
  switch(ProcStates) {
    case Idle:
      if (Input = LOW) {
        timer = millis();
        return Wait2;

      }
        break;
    case Wait2:
      if (Input = HIGH) {
        return Idle;

      }else if (timer - currenttime >= interval) {
       // Serial.println("indicator light is on");
        currenttime = timer;
        return Low2;

      }
        break;
   case Low2:
     if (Input = HIGH) {
        return Idle;

     }
  }
}

void loop(){
  NextState(ProcStates);

  Input = digitalRead(buttonPin);
  unsigned long currentTime = millis();

  if (Input != Wait){
    timeOfLastButtonEvent = currentTime;

  }

  if (currentTime - timeOfLastButtonEvent > debounceInterval) {                    //Checks on the voltage based on timer
    if (Input != Low) {                    //If the voltage has changed, switch states
      Low = Input;                                        //Updates the state

      //trigger an event
      if (Low == HIGH) {                                        //Does a serial print when button is pressed
        Serial.println("released");
        buttonPressCount++;
                          //Question 2, ads to counter when pressed and depressed
      } else {
        Serial.println("pressed");       //Serial print and adds to button press count
        buttonPressCount++; This is for part 1
      }
    }
  }
```