

```
#include <Keyboard.h>

bool comboMode = false;

// IR logic
bool irWasDetected = false;
bool irReadyForNextToggle = true;
unsigned long irDetectStart = 0;
unsigned long startupTime = 0;

// Puff logic
int puffCount = 0;
bool puffWasPressed = false;
unsigned long firstPuffTime = 0;

// Joystick key states
bool upPressed = false;
bool downPressed = false;
bool leftPressed = false;
bool rightPressed = false;

// Timing
const unsigned long IR_HOLD_TIME = 250;
const unsigned long BUTTON_DEBOUNCE = 200;
const unsigned long STARTUP_LOCKOUT = 1000;
const unsigned long DOUBLE_PUFF_WINDOW = 500;
const unsigned long PUFF_COOLDOWN = 150;

// Joystick thresholds
const int LOW_THRESHOLD = 300;
const int HIGH_THRESHOLD = 700;

// Pins
const int BUTTON_PIN = 2;
const int MICRO_PIN = 3;
const int IR_PIN = 4;
const int PUFF_PIN = 5;
const int BLUE_LED_PIN = 9;
const int GREEN_LED_PIN = 10;
const int JOY_X = A0;
const int JOY_Y = A1;
```

```

void updateLEDs() {
    if (comboMode) {
        digitalWrite(BLUE_LED_PIN, LOW);
        digitalWrite(GREEN_LED_PIN, HIGH);
    } else {
        digitalWrite(BLUE_LED_PIN, HIGH);
        digitalWrite(GREEN_LED_PIN, LOW);
    }
}

void setup() {
    pinMode(BUTTON_PIN, INPUT_PULLUP);
    pinMode(MICRO_PIN, INPUT_PULLUP);
    pinMode(PUFF_PIN, INPUT_PULLUP);
    pinMode(IR_PIN, INPUT);

    pinMode(BLUE_LED_PIN, OUTPUT);
    pinMode(GREEN_LED_PIN, OUTPUT);

    comboMode = false;
    updateLEDs();

    startupTime = millis();

    Serial.begin(9600);
    Keyboard.begin();
}

void loop() {
    bool buttonPressed = digitalRead(BUTTON_PIN) == LOW;
    bool microPressed = digitalRead(MICRO_PIN) == LOW;
    bool puffPressed = digitalRead(PUFF_PIN) == LOW;
    bool irDetected = digitalRead(IR_PIN) == LOW;

    // ----- IR MODE TOGGLE -----
    if (millis() - startupTime > STARTUP_LOCKOUT) {
        if (irDetected) {
            if (!irWasDetected) {
                irWasDetected = true;
                irDetectStart = millis();
            }
        }
    }
}

```

```

    if (irReadyForNextToggle && (millis() - irDetectStart >= IR_HOLD_TIME)) {
        comboMode = !comboMode;
        irReadyForNextToggle = false;
        updateLEDs();
        Serial.println(comboMode ? "COMBO MODE ON" : "COMBO MODE OFF");
    }
} else {
    irWasDetected = false;
    irReadyForNextToggle = true;
}
}

// ----- BUTTON -----
if (buttonPressed) {
    if (comboMode) {
        Keyboard.write('e');
        Serial.println("Combo Mode: Button = E");
    } else {
        Keyboard.write(' ');
        Serial.println("Normal Mode: Button = Space");
    }
    delay(BUTTON_DEBOUNCE);
}

// ----- MICROSWITCH -----
if (microPressed) {
    if (comboMode) {
        Keyboard.write('r');
        Serial.println("Combo Mode: Microswitch = R");
    } else {
        Keyboard.write('q');
        Serial.println("Normal Mode: Microswitch = Q");
    }
    delay(BUTTON_DEBOUNCE);
}

// ----- PUFF + DOUBLE PUFF -----
if (puffPressed && !puffWasPressed) {
    puffWasPressed = true;

    if (puffCount == 0) {
        puffCount = 1;
    }
}

```

```

        firstPuffTime = millis();
    } else if (puffCount == 1 && millis() - firstPuffTime <= DOUBLE_PUFF_WINDOW) {
        puffCount = 2;
    }

    delay(PUFF_COOLDOWN);
}

if (!puffPressed) {
    puffWasPressed = false;
}

if (puffCount == 1 && millis() - firstPuffTime > DOUBLE_PUFF_WINDOW) {
    if (comboMode) {
        Keyboard.write('f');
        Serial.println("Combo Mode: Single Puff = F");
    } else {
        Keyboard.write('c');
        Serial.println("Normal Mode: Single Puff = C");
    }
    puffCount = 0;
}

if (puffCount == 2) {
    if (comboMode) {
        Keyboard.write('x');
        Serial.println("Combo Mode: Double Puff = X");
    } else {
        Keyboard.write('z');
        Serial.println("Normal Mode: Double Puff = Z");
    }
    puffCount = 0;
}

// ----- JOYSTICK -----
int x = analogRead(JOY_X);
int y = analogRead(JOY_Y);

if (x < LOW_THRESHOLD) {
    if (!leftPressed) {
        Keyboard.press('a');
        leftPressed = true;
    }
}

```

```
    }  
    } else if (leftPressed) {  
        Keyboard.release('a');  
        leftPressed = false;  
    }  
  
    if (x > HIGH_THRESHOLD) {  
        if (!rightPressed) {  
            Keyboard.press('d');  
            rightPressed = true;  
        }  
    } else if (rightPressed) {  
        Keyboard.release('d');  
        rightPressed = false;  
    }  
  
    if (y < LOW_THRESHOLD) {  
        if (!upPressed) {  
            Keyboard.press('w');  
            upPressed = true;  
        }  
    } else if (upPressed) {  
        Keyboard.release('w');  
        upPressed = false;  
    }  
  
    if (y > HIGH_THRESHOLD) {  
        if (!downPressed) {  
            Keyboard.press('s');  
            downPressed = true;  
        }  
    } else if (downPressed) {  
        Keyboard.release('s');  
        downPressed = false;  
    }  
  
    delay(20);  
}
```