

```

#include <Arduino.h>

// ----- Pinos da IBT-2 -----
#define RPWM 25
#define LPWM 26
#define REN 27
#define LEN 14

// ----- Pinos dos Sensores -----
#define S_SUPERIOR 34 // topo
#define S_MEIO 35 // intermediário
#define S_INFERIOR 32 // base

char Rx;

// -----
// CONFIGURAÇÃO INICIAL DA PLACA + SEGURANÇA ANTI-BOOT
// -----
void ConfigPlaca() {

    // Desliga a ponte H durante a inicialização (evita motor ligar sozinho)
    digitalWrite(REN, 0);
    digitalWrite(LEN, 0);

    pinMode(RPWM, OUTPUT);
    pinMode(LPWM, OUTPUT);
    pinMode(REN, OUTPUT);
    pinMode(LEN, OUTPUT);

    // Pinos 34 e 35 não têm pull-up interno
    pinMode(S_SUPERIOR, INPUT);
    pinMode(S_MEIO, INPUT);

    // Pino 32 tem — deixa mais estável
    pinMode(S_INFERIOR, INPUT_PULLUP);

    // PWM seguro
    analogWrite(RPWM, 0);
    analogWrite(LPWM, 0);

    Serial.begin(9600);

    delay(300); // aguarda estabilização elétrica

    // Agora libera a ponte H
    digitalWrite(REN, 1);
    digitalWrite(LEN, 1);
}

```

```

// -----
// FUNÇÕES DO MOTOR
// -----

void motorUp() {
    analogWrite(RPWM, 127);
    analogWrite(LPWM, 0);
}

void motorDown() {
    analogWrite(RPWM, 0);
    analogWrite(LPWM, 127);
}

void motorStop() {
    analogWrite(RPWM, 0);
    analogWrite(LPWM, 0);
}

// -----
// CONTROLE DO MOTOR VIA SERIAL
// -----

void ControleMotor() {

    if (Serial.available() > 0) {
        Rx = Serial.read();

        switch (Rx) {

            case 'U': // SUBIR
                if (digitalRead(S_SUPERIOR) == HIGH) {
                    motorUp();
                    Serial.println("Subindo...");
                } else {
                    motorStop();
                    Serial.println("LIMITE SUPERIOR ATINGIDO");
                }
                break;

            case 'D': // DESCER
                if (digitalRead(S_INFERIOR) == HIGH) {
                    motorDown();
                    Serial.println("Descendo...");
                } else {
                    motorStop();
                    Serial.println("LIMITE INFERIOR ATINGIDO");
                }
                break;
        }
    }
}

```

```

    case 'S': // PARAR
        motorStop();
        Serial.println("Parado");
        break;
    }
}

// Failsafe automático
if (digitalRead(S_SUPERIOR) == LOW) motorStop();
if (digitalRead(S_INFERIOR) == LOW) motorStop();
}

// -----
void setup() {

    // --- ANTI-BOOT: garante motor OFF antes de tudo ---
    pinMode(RPWM, OUTPUT);
    pinMode(LPWM, OUTPUT);
    analogWrite(RPWM, 0);
    analogWrite(LPWM, 0);

    ConfigPlaca();
}

// -----
void loop() {
    ControleMotor();
}

```