```
/* KY040
  Forgás kódoló modul
  A forgatás bemutatása több módon történik.
  - Egy háromszínű LED modul az encoder lépéseit négyféle szín kapcsolgatásával mutatja be. Ha a forgásirányt
    megváltoztatjuk, a színek sorrendje is megfordul.
  - Két LED, egy piros és egy zöld mutatja a forgatás irányát.
  - A Soros portra kiírásra kerül a forgatás iránya és a számláló aktuális értéke.
  Bekötés:
    A háromszínű LED:
      piros: 2,
      zöld: 3,
      kék: 4.
    Az encoder:
      CLK: 6,
      DT:  7,
      SW:  5.
    A LED-ek:
      piros: 8,
      zöld:  9;
 */
#define CLK 6
#define DT 7
#define SW 5

int red = 2;
int green = 3;
int blue = 4;
int redLed = 8;
int greenLed = 9;
int counter = 0;
int currentStateCLK;
int lastStateCLK;
String currentDir ="";
unsigned long lastButtonPress = 0;
int rest = 0;

void setup() {
  // A kivezetések beállításai:
  pinMode(red, OUTPUT);
  pinMode(green, OUTPUT);
  pinMode(blue, OUTPUT);
  pinMode(redLed, OUTPUT);
  pinMode(greenLed, OUTPUT);
  pinMode(CLK, INPUT);
  pinMode(DT, INPUT);
  pinMode(SW, INPUT_PULLUP);
  setLights(LOW, LOW, LOW);
  // A Soros Monitor indítása
  Serial.begin(9600);

  // Olvassa el a CLK kezdeti állapotát
  lastStateCLK = digitalRead(CLK);

}

void loop() {
```

```
// Olvassa el a CLK aktuális állapotát
currentStateCLK = digitalRead(CLK);

// Ha a CLK utolsó és aktuális állapota különbözik, akkor impulzus történt
// Csak 1 állapotváltozásra reagáljon, hogy elkerülje a dupla számítást
if (currentStateCLK != lastStateCLK  && currentStateCLK == 1){

  // Ha a DT állapota eltér a CLK állapottól,
  // akkor a kódoló jobbra forog, így csökken
  if (digitalRead(DT) != currentStateCLK) {
    counter --;
    currentDir ="CCW";
  } else {
    // A kódoló CW-ra forog, így növekszik
    counter ++;
    currentDir ="CW ";
  }

  Serial.print("Direction: ");
  Serial.print(currentDir);
  Serial.print(" | Counter: ");
  Serial.println(counter);
  rest = counter % 4;
  if (rest < 0) rest += 4;
  switch (rest) {
    case 0:
      setLights(HIGH, LOW, HIGH);
      break;
    case 1:
      setLights(HIGH, LOW, LOW);
      break;
    case 2:
      setLights(LOW, HIGH, LOW);
      break;
    case 3:
      setLights(LOW, LOW, HIGH);
      break;
  }
  if (currentDir == "CCW") {
    digitalWrite(redLed, HIGH);
    digitalWrite(greenLed, LOW);
    }
  else {
    digitalWrite(greenLed, HIGH);
    digitalWrite(redLed, LOW);
  }
}

// Emlékezzen az utolsó CLK állapotra
lastStateCLK = currentStateCLK;

// A nyomógomb állapotának olvasása
int btnState = digitalRead(SW);

// Ha LOW jelet észlelünk, megnyomjuk a gombot
if (btnState == LOW) {
  // ha 50 ms telt el az utolsó LOW impulzus óta, az azt jelenti,
```

```
    // hogy a gombot megnyomták, felengedték és újra megnyomták
    if (millis() - lastButtonPress > 50) {
      Serial.println("Button pressed!");
    }

    // Emlékezzen az utolsó gombnyomás eseményére
    lastButtonPress = millis();
  }

  // Tegyen egy kis késleltetést az olvasás pergésmentesítéséért
  delay(1);
}

void setLights(int r, int g, int b) {
  digitalWrite(red, r);
  digitalWrite(green, g);
  digitalWrite(blue, b);
  }
```