

Sistemas con Microprocesadores

Práctica 5

Juan Antonio Martínez Sánchez

EJERCICIO:

```
#define PIN_INT_CODIF 2
```

```
#define PIN_POTEN A0
```

```
long Err_acum = 0;
```

```
uint8_t Rpm, Consigna_rpm;
```

```
unsigned long Tiempo_int_prev;
```

```
void rutina_int_codif()
```

```
{
```

```
    unsigned long tiempo_int_actual = micros();
```

```
    unsigned long periodo = tiempo_int_actual - Tiempo_int_prev;
```

```
    Rpm = 60*1000000 / periodo / 1656;
```

```
    Tiempo_int_prev = tiempo_int_actual;
```

```
}
```

```
uint8_t computa_i() {
```

```
    const int ki = 10;
```

```
    int pwm_inicial;
```

```
    uint8_t pwm_final;
```

```
    int err, compon_ki;
```

```
    static unsigned long tiempo_i_prev = 0; // Con static se conserva el valor de la variable de una llamada de la fn a otra
```

```
    unsigned long tiempo_i_act = millis();
```

```
    unsigned long tiempo_transcurr_i = tiempo_i_act - tiempo_i_prev;
```

```
    err = Consigna_rpm - Rpm;
```

```
    Err_acum += err * tiempo_transcurr_i;
```

```
    compon_ki = ki * Err_acum/1000;
```

```
    pwm_inicial = compon_ki;
```

```
    if(pwm_inicial > 255)
```

```
        pwm_final = 255;
```

```
    else if(pwm_inicial < 1)
```

```
        pwm_final = 1; // No detenemos completamente el motor nunca para no dejar de generar interrupciones
```

```
    else
```

```
        pwm_final = pwm_inicial;
```

```
    tiempo_i_prev = tiempo_i_act;
```

```
    return pwm_final;
```

```
}
```

```

void setup()
{
  Serial.begin(9600);
  pinMode(PIN_INT_CODIF, INPUT_PULLUP);
  pinMode(PIN_POTEN, INPUT);
  pinMode(3, OUTPUT);
  digitalWrite(3, 0);

  rutina_int_codif();

  attachInterrupt(digitalPinToInterrupt(PIN_INT_CODIF),rutina_int_codif, RISING);
}

void loop()
{
  uint16_t volt_poten = analogRead(PIN_POTEN);

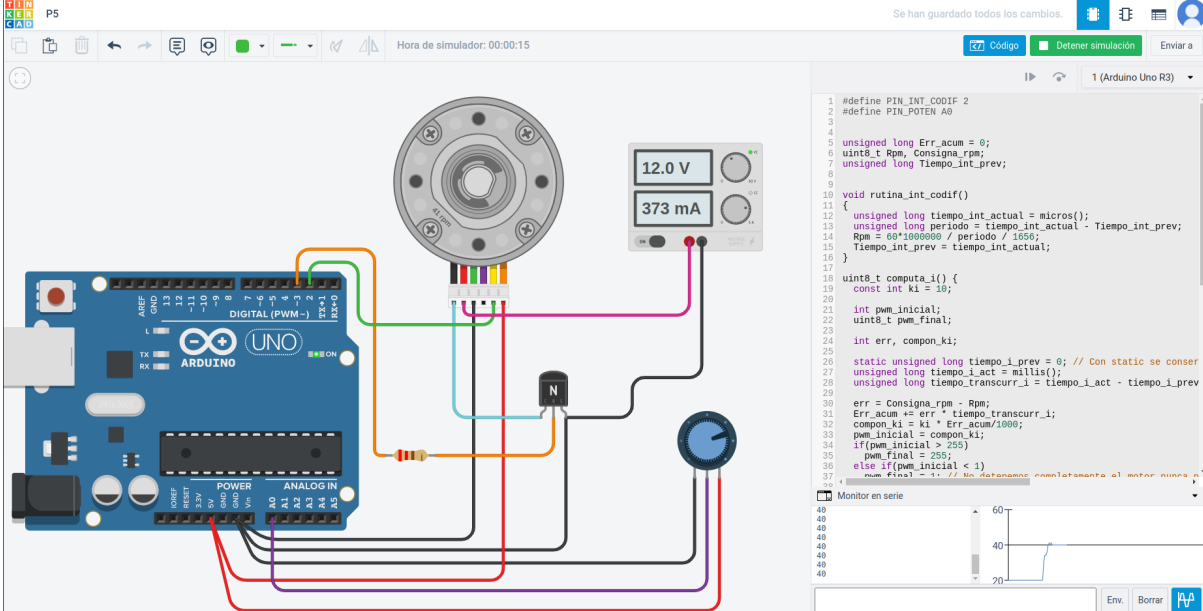
  Consigna_rpm = map(volt_poten, 0, 1023, 10, 50);
  Serial.println(Consigna_rpm);

  uint8_t pwm = computa_i();
  analogWrite(3, pwm);

  delay(100);
}

```

Motor a 40 R.P.M.:



The screenshot displays the Arduino IDE interface. On the left, a circuit diagram shows an Arduino Uno connected to a motor. The motor is powered by a 12.0V source and its current is measured at 373 mA. The motor's speed is set to 40 RPM. The code on the right defines the pins for the motor's control and the potentiometer. The code includes a setup function to initialize the serial port and the motor's control pins, and a loop function that reads the potentiometer's value, maps it to a target RPM, and then calculates the PWM duty cycle to control the motor's speed. The code also includes a delay of 100ms between each loop iteration.

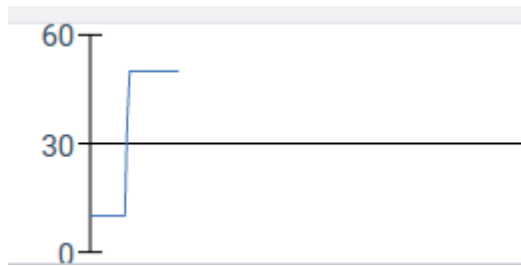
```

1 #define PIN_INT_CODIF 2
2 #define PIN_POTEN A0
3
4
5 unsigned long Err_acum = 0;
6 uint8_t Rpm, Consigna_rpm;
7 unsigned long Tiempo_int_prev;
8
9
10 void rutina_int_codif()
11 {
12   unsigned long tiempo_int_actual = micros();
13   unsigned long periodo = tiempo_int_actual - Tiempo_int_prev;
14   Rpm = 60*1000000 / periodo / 1056;
15   Tiempo_int_prev = tiempo_int_actual;
16 }
17
18 uint8_t computa_i() {
19   const int ki = 10;
20
21   int pwm_inicial;
22   uint8_t pwm_final;
23
24   int err, compon_ki;
25
26   static unsigned long tiempo_i_prev = 0; // Con static se conser
27   unsigned long tiempo_i_act = millis();
28   unsigned long tiempo_transcurr_i = tiempo_i_act - tiempo_i_prev
29
30   err = Consigna_rpm - Rpm;
31   Err_acum += err * tiempo_transcurr_i;
32   compon_ki = ki * Err_acum/1000;
33   pwm_inicial = compon_ki;
34   if(pwm_inicial > 255)
35     pwm_final = 255;
36   else if(pwm_inicial < 1)
37     pwm_final = 1; // En defecto ralentiza el motor a 10 RPM
38   return pwm_final;
39 }

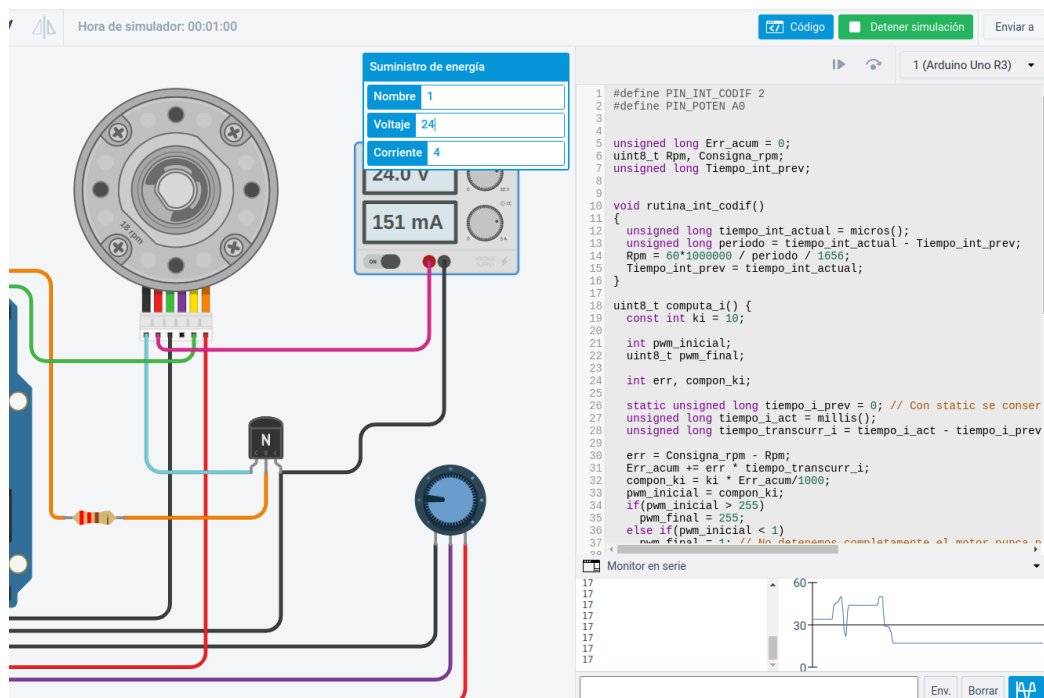
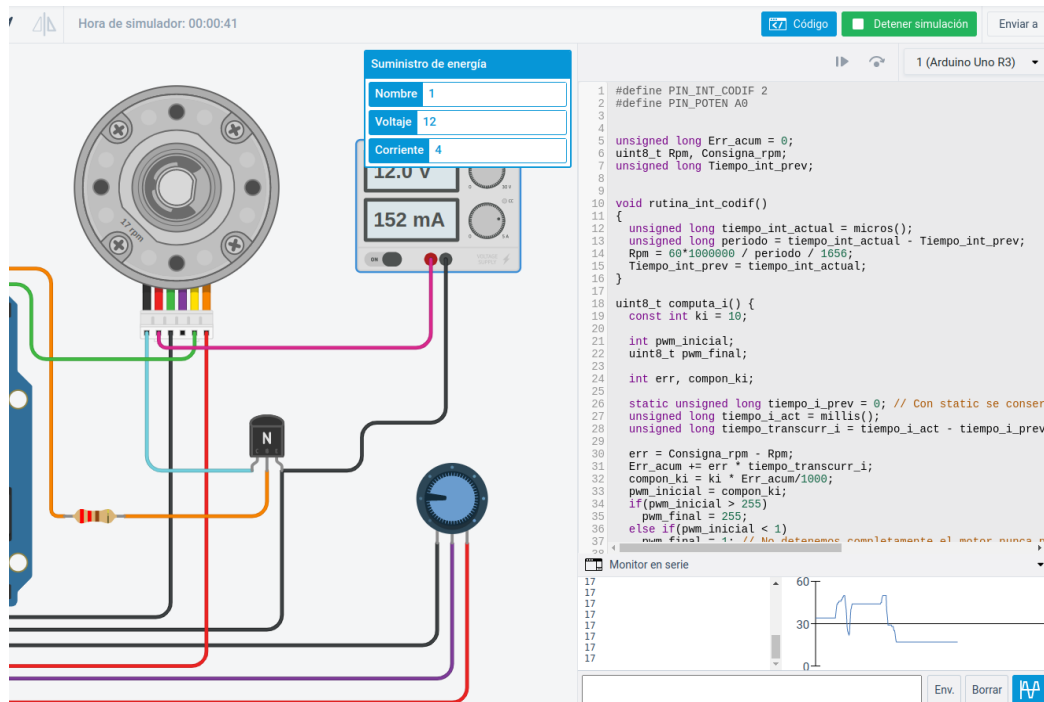
```

The monitor window shows the output of the serial port, displaying the target RPM (Consigna_rpm) and the calculated PWM duty cycle (pwm).

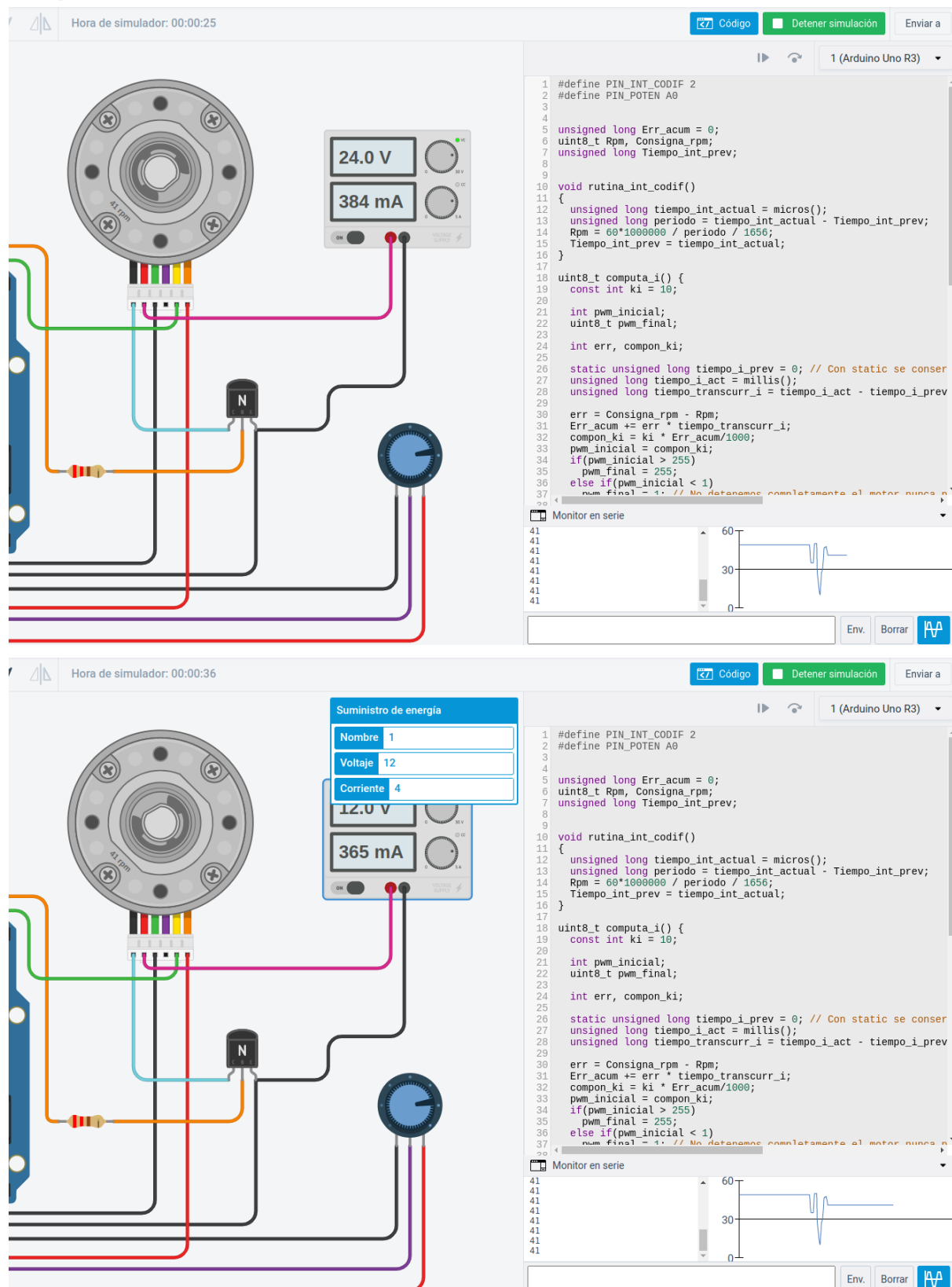
Adaptación a un cambio de consigna de 10 a 50 R.P.M.:



Adaptación a un cambio de tensión de la fuente de 12 V a 24 V:



Adaptación a un cambio de 24 V a 12 V:



En ambos casos se adapta al cambio brusco de tensión manteniendo constantes las R.P.M.