Annex (A)

```
 Timer1 library example
 *  June 2008 | jesse dot tane at gmail dot com
 */

#include "TimerOne.h"
#include <math.h>
#include <PID_v1.h>

// Parametre du regulateur
#define Kp -0.03
#define Ki -0.2
#define Kd 0
#define Kpi 30
#define Kii 2
#define Kdi 0
// set reference
//#define V -5

int commande;
float  V,tension, I, erreur, Somme_erreur, variation_erreur, erreur_precedente, commandeP ;
double Setpoint, Input, Output, I_ref,erreur_i,Somme_erreur_i, variation_erreur_i,
erreur_i_precedente;
void setup()
{
  Timer1.initialize(32);        // initialize timer1, and set a 1/2 second period
  Timer1.pwm(9, 1000);              // setup pwm on pin 9, 50% duty cycle
  Timer1.attachInterrupt(callback);  // attaches callback() as a timer overflow interrupt

  Serial.begin(115000);
  // Serial.println("--- Start Serial Monitor SEND_RCVE ---");
```

```
  //Serial.println(" Type in Box above, . ");
  //Serial.println("(Decimal)(Hex)(Character)");
  //Serial.println();
}

void callback()
{
  if (commande < 0) {
    commande = 0 ;
  }

  else if (commande > 1020) {
    commande = 1020;
  }

  Timer1.setPwmDuty(9,commande);
}



void loop()
{
  Setpoint=-8;
  I=analogRead(A0);
  I=(0.0264*I-13.5611);
  Input = analogRead(A1);

  Input = -(Input* 0.4883) ;
 erreur =Setpoint - Input ;
  Somme_erreur += erreur ;
  variation_erreur = erreur - erreur_precedente ;
 I_ref= Kp * erreur + Ki * Somme_erreur + Kd * variation_erreur ;
```

```
 //I_ref=0.5;
 erreur_i=I_ref-I;
 Somme_erreur_i += erreur_i ;
  variation_erreur_i = erreur_i - erreur_i_precedente ;
 commande= Kpi * erreur_i + Kii * Somme_erreur_i + Kdi * variation_erreur_i ;
 //commande=0.00238*commandeP+29.58*erreur-25.02*erreur_precedente;
  erreur_precedente = erreur ;
  erreur_i_precedente = erreur_i;

}
```