

## Commande de ventilateur

Destiné à refroidir le transistor de sortie d'une alimentation de labo programmable Tuxgraphics.

Je lui demande de démarrer le ventilateur vers 30-35°C, puis une zone proportionnelle jusqu'à la température max au-delà de laquelle le ventilateur est à fond. Un dégomme sera fait à la mise sous tension.

Ventilateur 60 x 60 12V 0,2A sur radiateur 60 x 60 x40 (récupéré d'un ancien PC).

## Architecture

- un microcontrôleur atmel AT tiny13 (à 0,20\$)
- un MOSFET BS170 pour commander le ventilateur (à 0,1\$)
- une thermistance 10k (à partir de 0,06\$)
- comme c'est un pilotage à  $\mu\text{C}$ , on peut l'adapter à n'importe quel capteur y compris le DS18B20 à 2\$, bien plus cher qu'une thermistance, mais à la mode, bien que sa précision soit inutile ici

## Thermistance

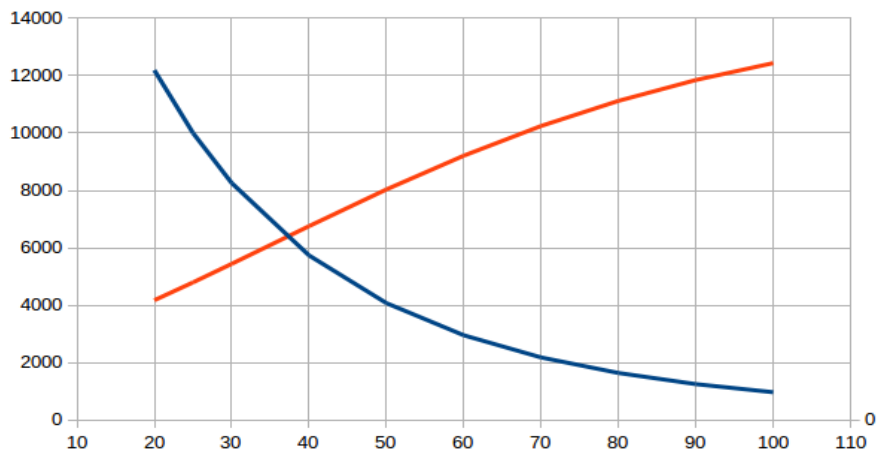
Etalonnée en trois points : 1,3°C=12189, 23°C=10920, 49°C=4320

puis extrapolation de la courbe

<http://www.thinksrs.com/downloads/programs/Therm%20Calc/NTCCalibrator/NTCcalculator.htm>

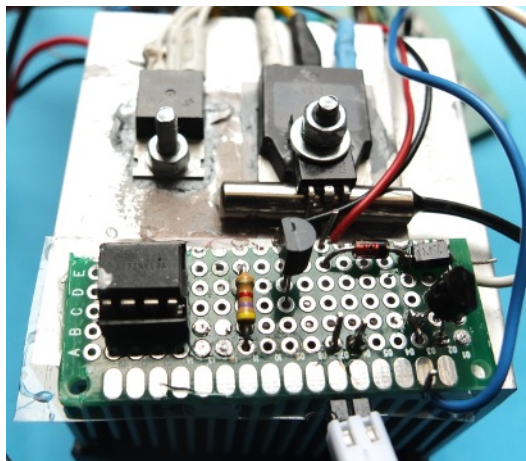
on en extrait la table, le  $\beta$  (3590) et la figure suivante (la courbe rouge de tension est multipliée par x, juste pour remplir correctement la figure)

temp	Rth	V $\mu\text{C}$
2,9	26200	0,76
25	10000	1,60
30	8257	1,81
40	5735	2,25
50	4073	2,68
60	2954	3,07
70	2182	3,41
80	1640	3,71
90	1252	3,95
100	970	4,14



L'entrée analogique ADC est en pin 2 = PB3. La thermistance est reliée au +5V, une résistance de 4,7 k $\Omega$  à la masse, ainsi lorsque la température monte, la tension monte aussi. Je préfère dans ce sens, juste pour mon confort intellectuel. Un LM78L05 sert de régulateur pour le  $\mu\text{C}$  et de référence de tension.

Le capteur (dans un tube inox) est pincé contre le transistor de puissance, avec adjonction de graisse thermique

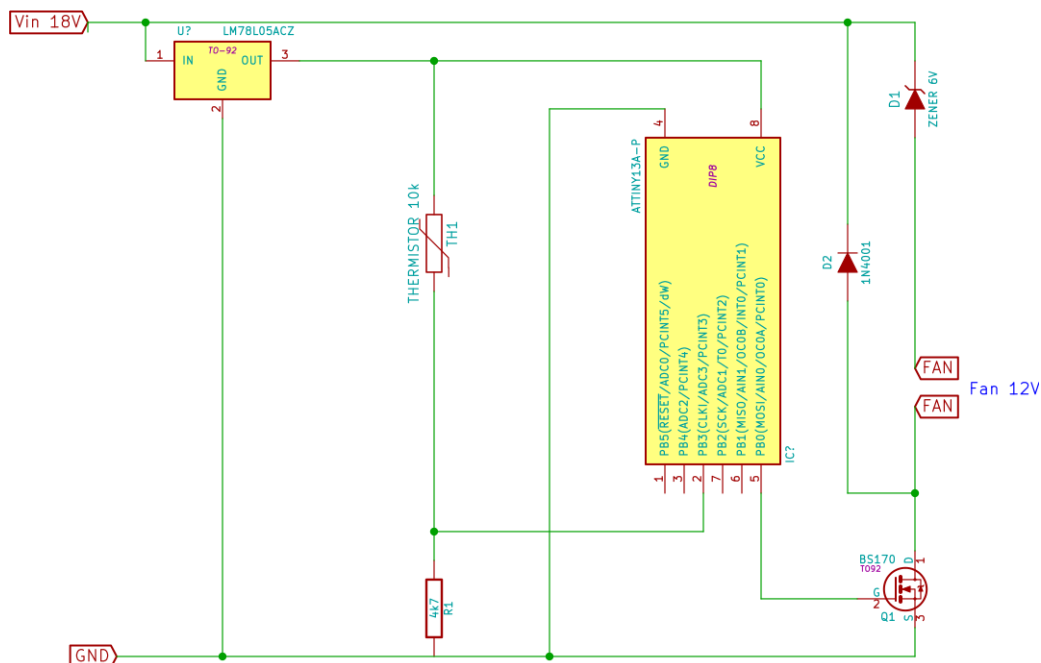


La zone proportionnelle commence vers 30°C et le ventilateur est à fond à partir d'environ 50°C.

## Schéma

KISS (keep it simple stupid)

montage réalisé pour une tension d'entrée de l'alimentation de labo de 18V (un bloc d'alim de PC portable). Facilement adaptable à une autre tension : la Zener absorbe la différence de tension entre la tension d'entrée (qui doit être supérieure à 7V) et la tension du ventilateur.



## Programme :

```
/* at tiny13
 *
 * fan automatic control
 * proportional control from 40°C to 70°C
 * hfuse FF lfuse 6A
 *
 * Zibuth27 2015/10/24
 *
 * PB3 pin2 = ADC3 = thermistor to +5, 4k7 to GND
 * PB0 pin5 = out PWM
 *
 * starts around 40°C
 * proportional up to 80°C
 * full blow above
 *
 * status: OK
 * keywords: ADC, PWM
 */

#define F_CPU 1200000UL // 1.2 MHz
#include <util/delay.h>
#define TRIGG_TEMP 370
#include <avr/io.h>
#include <avr/sfr_defs.h>

void main()
{
    uint8_t i, cmd;
    uint16_t temperature;

    DDRB = 0x11; // output OC1A enable
```

```

TCCR0A = 0x81;
TCCR0B = 0x03;

ADMUX = 0x03 ;           // mux0 PC0
ADCSRA = 0x85;           // enable ADC, prescaler 1024

// initial check : correct running of fan

i=0;
while (i<255)
{
    OCR0A = i;
    _delay_ms(20);
    i++;
}
_delay_ms(3000);
while(i>0)
{
    OCR0A = i;
    _delay_ms(20);
    i--;
}
OCR0A = 0;

_delay_ms(3000);

while (1)                // endless
{
// temperature capture
    ADCSRA |= (1<<ADSC);
    while (ADCSRA & (1<<ADSC));
    temperature = ADC;

// fan control
    if( temperature < TRIGG_TEMP) OCR0A = 0;
    if (temperature >TRIGG_TEMP)
    {
        cmd = temperature-TRIGG_TEMP;
        OCR0A = cmd;
    }
    if (temperature > (TRIGG_TEMP + 254)) OCR0A = 254;

    _delay_ms(500);

}
return 0;
}

```

fichier format Intel, à enregistrer sous la forme \*.hex, directement programmable :

```

:1000000009C00EC00DC00CC00BC00AC009C008C09A
:1000100007C006C011241FBECFE9CDBF02D052C019
:10002000EFCF81E187BB81E88FBD83E083BF87B9D4
:1000300085E886B980E086BFEE6F7E13197F1F712
:1000400000C000008F5F8F3FB1F7FFE72CEF3AE071
:10005000F15020403040E1F700C0000086BFEE6DD
:10006000F7E13197F1F700C000008150B9F716BEF3
:10007000FFE72CEF3AE0F15020403040E1F700C0BC
:1000800000002EEF369A3699FECF84B195B18237B3
:10009000E1E09E0710F416BE0BC08237F1E09F0727
:1000A00039F03EE8380F36BF8137924008F026BF5E
:1000B0008FEB94EDE1E081509040E040E1F700C02B
:0800C0000000E0CFF894FFCF2F
:00000001FF

```