

Подключение датчика KTS-1 к входу АЦП

При подключении датчика через вход АЦП потребуется использование дополнительного схемного элемента - резистора. Также потребуется рассчитать значение температуры по показаниям АЦП используя специальную формулу, представленную ниже.

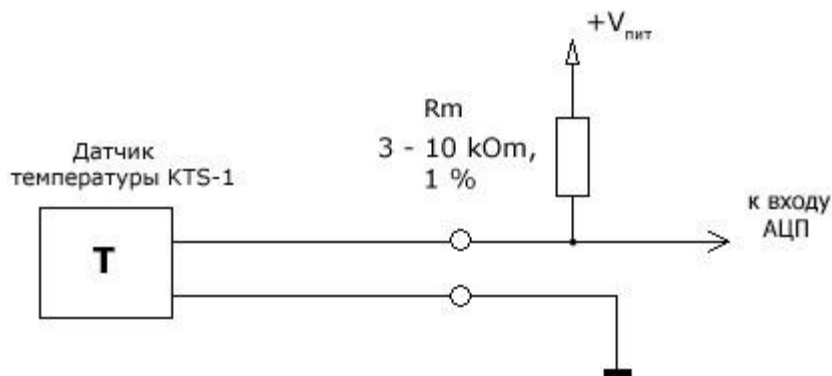


Рис.1 Схема подключения датчика температуры KTS-1 к входу АЦП

Величина сопротивления R_m , показанного на схеме ниже: порядка 3 кОм, хотя вполне допустимо выбрать любое доступное сопротивление из диапазона 2 - 10 кОм. Настоятельно рекомендуется использовать прецизионное сопротивление с точностью 1 %. Порядок (полярность) подключения выводов датчика KTS-1 к входу АЦП значения не имеет. $V_{пит}$ - источник постоянного опорного напряжения. Его величина не должна выходить за границы допустимого входного напряжения для АЦП конкретного модуля.

Для расчета температуры датчика в градусах Цельсия необходимо воспользоваться формулой, приведенной ниже.

$$R_t = \frac{R_m * V_{adc}}{V_{пит} - V_{adc}} \quad [Ом]$$

$$T = \frac{\beta}{\ln\left(\frac{R_t}{R_0}\right) + \frac{\beta}{298}} - 273 \quad [C^0]$$

где:

V_{adc} - показания АЦП в вольтах, к которому подключен датчик
 β - коэффициент температурной чувствительности датчика, равен 3988
 R_0 - сопротивление датчика при 25 °С, равно 3 кОм

Рассмотрим пример подключения датчика KTS-1 к модулю [MP714](#). На рисунке ниже представлена схема подключения датчика к модулю:

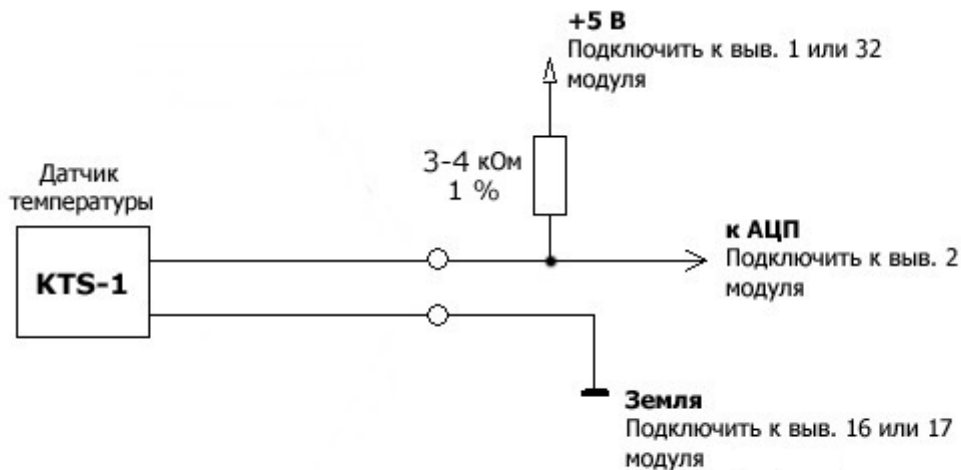


Рис.2 Схема подключения датчика температуры KTS-1 к модулю MP714

В качестве источника опорного напряжения для датчика выбрано напряжение питания USB величиной 5 В. Ниже представлен листинг простой консольной программы, написанной на языке C++, иллюстрирующий процесс расчета значения температуры по показаниям АЦП. В данном примере значение сопротивления R_m выбрано равным 4.3 кОм.

```
#include<iostream>
#include "windows.h"
#include<conio.h>
#include<math.h>

// Параметры датчика и схемы его подключения
doublebeta = 3988;
doubleRm = 4300;
double R0 = 3000;

int main(intargc, char *argv[])
{
if(argc< 2 )
{
printf("Error! No arguments.");
return 0;
}

// Входной аргумент - цифровое значение АЦП в диапазоне 0-1023
intADC Value = atoi(argv[1]);

// Расчет напряжения АЦП в вольтах
doubleinVolt = (((double)ADC Value)/1023.0)*5.09; // 5.09 - см.
Дополнения ниже
printf("InVolt: %f\r\n", inVolt);
```

```

// Расчет сопротивления терморезистора
double RT = (Rm*inVolt)/(5.0 - inVolt);
printf("RT: %f\r\n", RT);

// Расчет температуры в градусах Цельсия
double Temper = (beta/(log(RT/R0) + beta/298.0)) - 273;
printf("Temperature: %f\r\n", Temper);

return 0;
}

```

Дополнения:

- В качестве R_m рекомендуется применять прецизионный резистор с точностью 1 %. Использование резистора обычной штатной точности (5%) может оказать ощутимое влияние на результирующую точность измерений температуры. Ошибка в таком случае может достигать 1-2 °С.
- При использовании шины USB в качестве источника питания цепи термодатчика необходимо обратить внимание на тот факт, что номинально напряжения питания USB не равно 5 В а несколько больше или меньше этой величины. Обычно оно составляет 5.09 В для PC с внешним питанием и 4.8 - 4.9 В для ноутбука, работающего от внутренней батареи. Рекомендуется учитывать это в формуле для расчета напряжения АЦП. Пренебрежение этой особенностью может повлиять на точность измерения температуры (ошибка может достигать 0.5 - 1.0 °С)