

г. Караганда, ул. Алиханова 37, офис 108
г. Астана, ул. Ауэзова, 33/1, офис 210

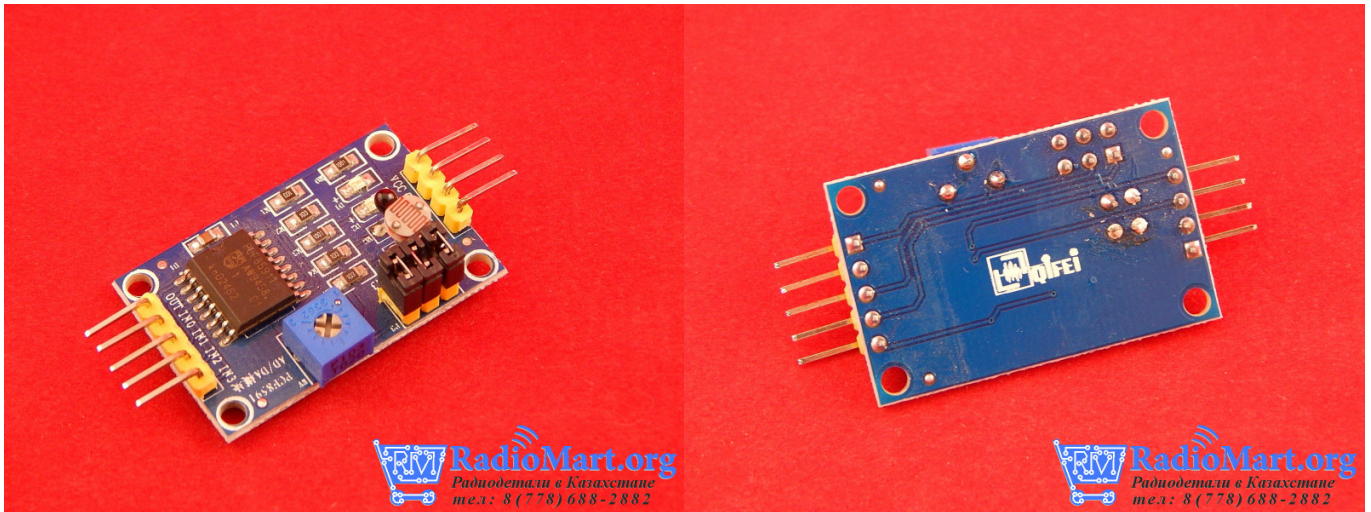
E-Mail: support@radiomart.org



Артикул: 10933

Цена в прайсе: 992 тг.

Преобразователь АЦП/ЦАП PCF8591



Конвертер построенный на базе микросхемы PCF8591, имеет 4 аналоговых входа (АЦП, он же ADC), 1 аналоговый выход (ЦАП, он же DAC), разрешение каждого входа/выхода 8 бит. Управляется по шине I2C с помощью Ардуино, Raspberry Pi или другого контроллера поддерживающего этот протокол. Рабочее напряжение 2.5 - 6В.

Можно использовать, если не хватает аналоговых входов/выходов на микроконтроллере - вешаем плату на шину I2C (она же IIC или TWI) и получаем данные с датчиков. Если сравнивать с АЦП Ардуино, то данный модуль проигрывает по разрешению (у Ардуинового АЦП 10-битное разрешение), зато можно сравнивать не просто поступающее напряжение как в Ардуино (от 0 и до 5В), а сравнивать между собой напряжения из разных источников (например для контроля уровня батарей), когда на один вход подается напряжение с одного источника, на второй - с другого и сравниваем результат (подробней можно почитать в даташите). Так же, можно измерить напряжение на всех 4 входах одновременно и затем получить 4 байта со значением по каждому входу (Ардуино так тоже не умеет).

Пример использования на Arduino

```
#include "Wire.h"
#define PCF8591 (0x90 >> 1) /* I2C адрес, может отличаться от вашего,
запустите скетч I2Cscanner
из примеров библиотеки Wire, чтобы получить правильный адрес*/
void setup(){
  Wire.begin();
```

```

}
void loop(){
  for (int i=0; i<256; i++) { //в цикле увеличиваем напряжение на AOUT
    Wire.beginTransmission(PCF8591); // начинаем передачу PCF8591
    Wire.write(0x40); // управляющая команда включающая ЦАП (бинарное 1000000)
    Wire.write(i); // значение, которое будет выдано на AOUT
    Wire.endTransmission(); // закончили передачу
  }
  for (int i=255; i>=0; --i) { //в цикле уменьшаем напряжение на AOUT
    Wire.beginTransmission(PCF8591);
    Wire.write(0x40);
    Wire.write(i);
    Wire.endTransmission();
  }
}
}

```

Выводим значение на всех аналоговых входах в терминал на Arduino

```

#include "Wire.h"
#define PCF8591 (0x90 >> 1) // I2C адрес
#define ADC0 0x00 // байт, где хранится адрес первого входа
#define ADC1 0x01 // байт, где хранится адрес второго входа
#define ADC2 0x02 // и так далее
#define ADC3 0x03
byte value0, value1, value2, value3;
void setup(){
  Wire.begin();
  Serial.begin(9600);
}
void loop(){
  Wire.beginTransmission(PCF8591); // начинаем передачу
  Wire.write(ADC0); // сообщаем модулю, что нас интересует первый вход
  Wire.endTransmission(); // завершаем передачу
  Wire.requestFrom(PCF8591, 2); //запрашиваем два байта от модуля
  value0=Wire.read(); //в первом байте будет прошлое значение датчика, оно нам
не надо
  value0=Wire.read(); //а вот во втором текущее, его и будем выводить
  Wire.beginTransmission(PCF8591); // повторяем операцию для остальных входов
  Wire.write(ADC1);
  Wire.endTransmission();
  Wire.requestFrom(PCF8591, 2);
  value1=Wire.read();
  value1=Wire.read();
  Wire.beginTransmission(PCF8591);
  Wire.write(ADC2);
  Wire.endTransmission();
  Wire.requestFrom(PCF8591, 2);
  value2=Wire.read();
  value2=Wire.read();
  Wire.beginTransmission(PCF8591);
  Wire.write(ADC3);

```

```
Wire.endTransmission();  
Wire.requestFrom(PCF8591, 2);  
value3=Wire.read();  
value3=Wire.read();  
Serial.print(value0); Serial.print(" "); //выводим значения в монитор  
Serial.print(value1); Serial.print(" ");  
Serial.print(value2); Serial.print(" ");  
Serial.print(value3); Serial.print(" ");  
Serial.println();  
delay(200); //делаем паузу, чтоб не мельтешило  
}
```