# 17.PCF8591_AD_Converter

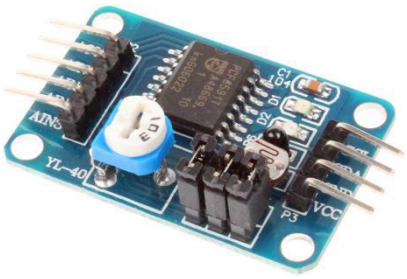
## Introduction

The PCF8591 is a single-chip, single-supply low-power 8-bit CMOS data acquisition device with four analog inputs, one analog output and a serial I2C-bus interface. Three address pins A0, A1 and A2 are used for programming the hardware address, allowing the use of up to eight devices connected to the I2C-bus without additional hardware. Address, control and data to and from the device are transferred serially via the two-line bidirectional I2C-bus.

The functions of the device include analog input multiplexing, on-chip track and hold function, 8-bit analog-to-digital conversion and an 8-bit digital-to-analog conversion. The maximum conversion rate is given by the maximum speed of the I2C-bus.

## Hardware Required

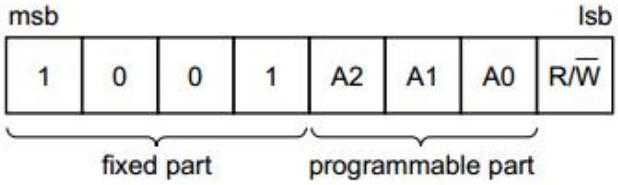
- ✓ 1 * Raspberry Pi
- ✓ 1 * Breadboard
- ✓ 1 * Network cable (or USB wireless network adapter)
- ✓ 1 * PCF8591 module
- ✓ 1 * Dual-Color LED module
- ✓ 1 * 3-Pin anti-reverse cable
- ✓ Several Jumper wires (M to F)




## Principle

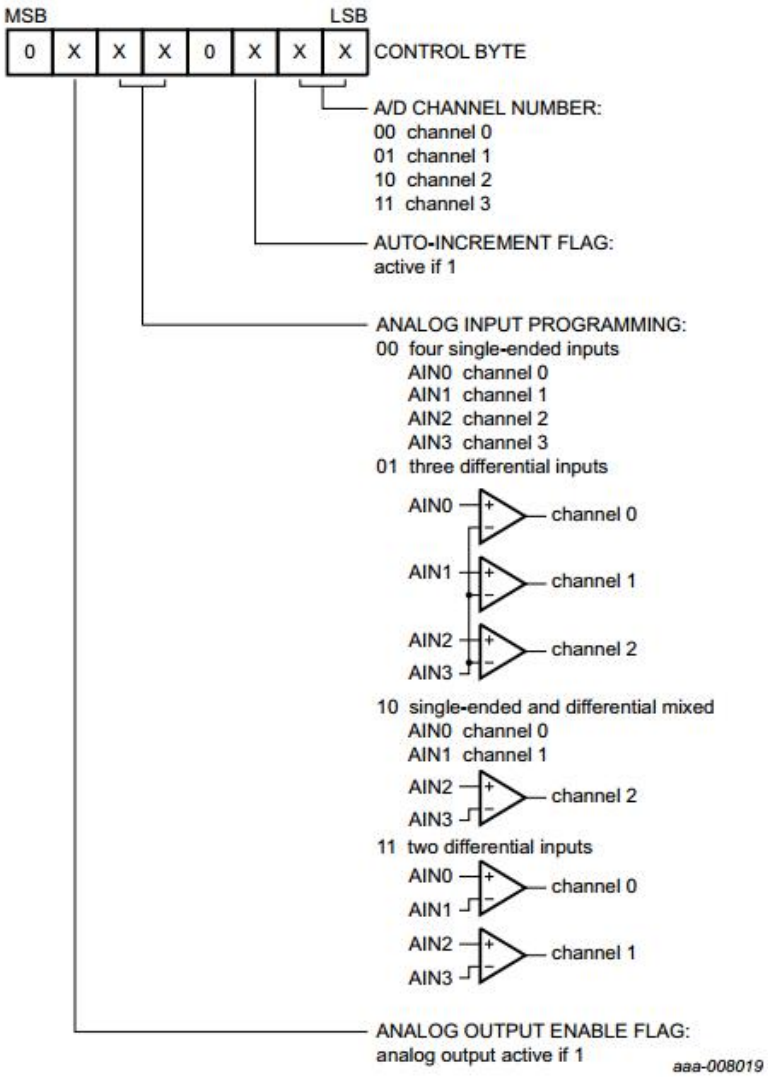
**Addressing:**

Each PCF8591 device in an I2C-bus system is activated by sending a valid address to the device. The address consists of a fixed part and a programmable part. The programmable part must be set according to the address pins A0, A1 and A2. The address always has to be sent as the first byte after the start condition in the I2C-bus protocol. The last bit of the address byte is the read/write-bit which sets the direction of the following data transfer (see the figure below).




**Control byte:**

The second byte sent to a PCF8591 device will be stored in its control register and is required to control the device function. The upper nibble of the control register is used for enabling the analog output, and for programming the analog inputs as single-ended or differential inputs. The lower nibble selects one of the analog input channels defined by the upper nibble (see Fig.5). If the auto-increment flag is set, the channel number is incremented automatically after each A/D conversion. See the figure below.



# 17.PCF8591_AD_Converter

In this experiment, the AIN0 (Analog Input 0) port is used to receive analog signals from the potentiometer module, and AOUT (Analog Output) is used to output analog signals to the dual-color LED module so as to change the luminance of the LED.

The PCF8591 is a monolithically integrated, individually powered, low power, 8-bit CMOS data acquisition device. The PCF8591 has four analog inputs, one analog output and one serial I2C bus interface. The three address pins A0, A1 and A2 of the PCF8591 can be used for hardware address programming, allowing access to eight PCF8591 devices on the same I2C bus without the need for additional hardware. In the PCF8591 device input and output address, control and data signals are through the two-way two-way I2C bus to serial transmission.

PCF8591 main performance indicators:

★ Single power supply

★ PCF8591 operating voltage range of 2.5V-6V

★ Low standby current

Serial input / output via I2C bus

The PCF8591 is addressed by three hardware address pins

The sampling rate of the PCF8591 is determined by the I2C bus speed

★ 4 analog inputs can be programmed as single-ended or differential inputs

★ Automatic incremental channel selection

★ The analog voltage range of the PCF8591 is from VSS to VDD

★ PCF8591 built-in track and hold circuit

★ 8-bit successive approximation A / D converter

★ DAC gain is achieved by one analog output

**Two baby description**

1 module chip using PCF8951

2 module supports external 4-channel voltage input acquisition (voltage input range 0-5v)

3 module integrated photoresistor, the ambient light can be collected by AD accurate value

4 module integrated thermistor, can be collected through the AD accurate temperature

value of the environment

5 module integrated 1 0-5V voltage input acquisition (through the blue potentiometer to adjust the input voltage)

6 module with power indicator (indicator lights when power is supplied to the module)

7 module with DA output indicator, when the module DA output interface voltage reaches a certain value, will light up the board DA output indicator, the greater the voltage, the greater the brightness of the indicator light;

8 module PCB size: 3.6cm * 2.3cm

9 standard double-sided board, thickness 1.6mm, beautiful and generous layout, surrounded by a through-hole, aperture: 3mm, easy to fix

**Three module interface description**

The left and right side of the module, respectively, outside the expansion of 2-pin connector, respectively, as follows:

Left AOUT chip DA output interface

AINO chip analog input interface 0

AIN1 chip analog input interface 1

AIN2 chip analog input interface 2

AIN3 chip analog input interface 3

SCL IIC clock interface on the right then single-chip IO port

SDA IIC digital interface connected MCU IO port

GND module to ground

VCC power interface external 3.3v-5v

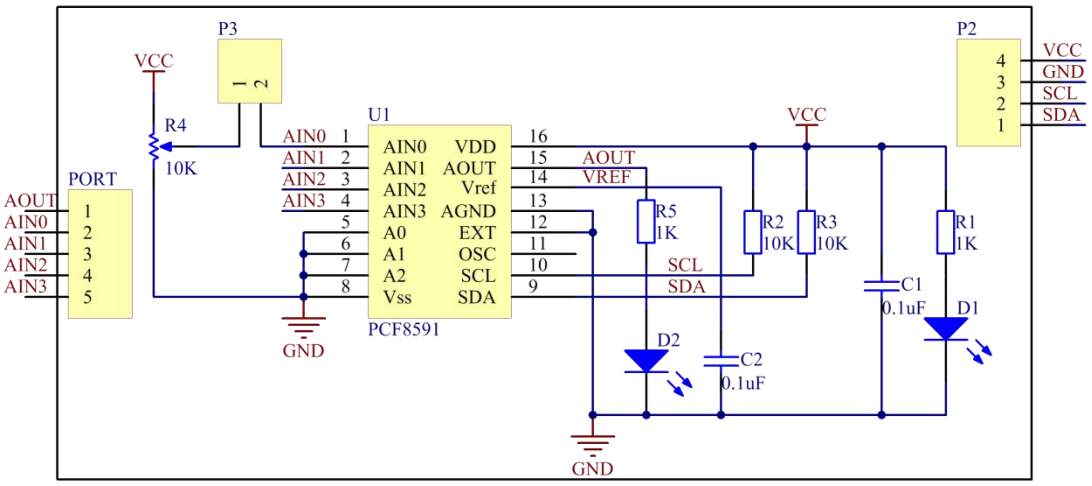
Four modules red short-circuit cap instructions

Module a total of three red short-circuit cap, respectively, as follows:

P4 connected to P4 short-circuit cap, select the thermistor access circuit

P5 connected to P5 short-circuit cap, select the photosensitive resistor access circuit

P6 connected to P6 short-circuit cap, select 0-5V adjustable voltage access circuit




## Schematic Diagram

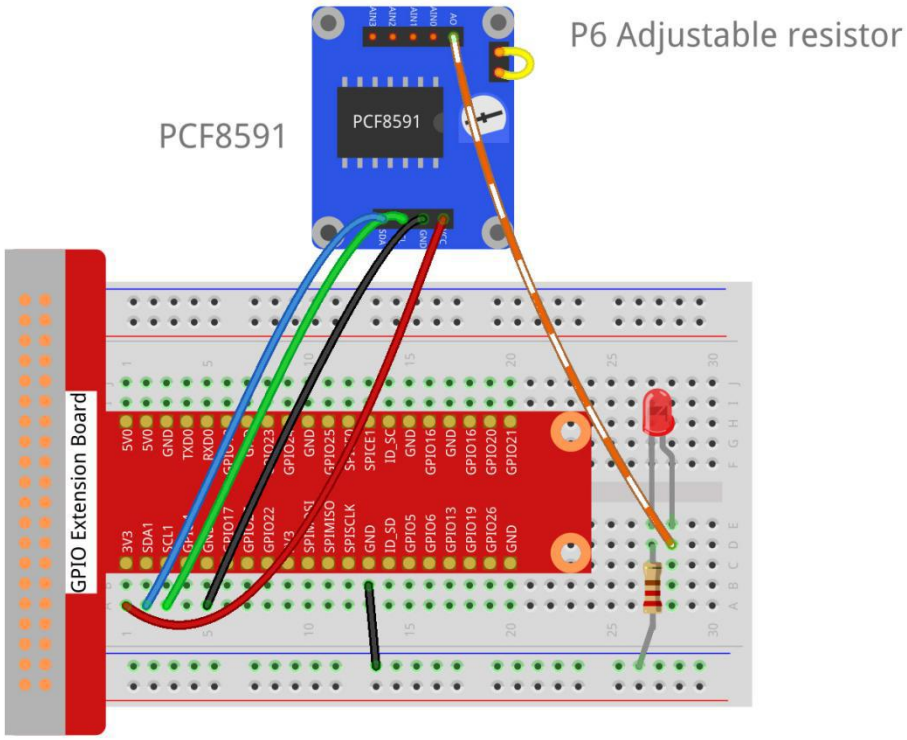
| Raspberry Pi | PCF8591 Module |
|---|---|
| SDA | SDA |
| SCL | SCL |
| 3V3 | VCC |
| GND | GND |

| Dual-Color Module | PCF8591 Module |
|---|---|
| R | AOUT |
| GND | GND |
| G | * |

## Experimental Procedures

### Step 1: Build the circuit

Note:Connect the two pins next to the potentiometer of the PCF8591 module with the yellow jumper cap attached.




## Step 2: Setup I2C (see Appendix 1. If you have set I2C, you can skip this step.)

**For C language users:**

## Step 2: Go to the folder of the code.

```
cd /home/pi/REXQualis_Raspberry_Pi_Complete_Starter_Kit/C/17.PCF8591
```

## Step 3: Compile the code

```
gcc 17.PCF8591.c -o PCF8591.out -lwiringPi
```

## Step 4: Run the executable file output in the previous step.

```
sudo ./PCF8591.out
```

## Code

```
#include <stdio.h>
#include <wiringPi.h>
#include <pcf8591.h>
```

```
#define PCF             120


int main (void)
{
    int value ;
    wiringPiSetup () ;
    // Setup pcf8591 on base pin 120, and address 0x48
    pcf8591Setup (PCF, 0x48) ;
    while(1) // loop forever
    {
        value = analogRead    (PCF + 0) ;
        printf("%d\n", value);
        analogWrite (PCF + 0, value) ;
        delay (10) ;
    }
    return 0 ;
}
```

**For Python users:**

### Step 2: Go to the folder of the code and run it.

```
cd /home/pi/REXQualis_Raspberry_Pi_Complete_Starter_Kit/Python
```

### Step 3: Run the code

```
sudo python3 17.PCF8591.py
```

### Code

The code here is for Python3, if you need for Python2, please open the code with the suffix py2 in the attachment.

```
#!/usr/bin/env python3
import PCF8591 as ADC
```

```
def setup():
    ADC.setup(0x48)

def loop():
    while True:
        print (ADC.read(0))
        ADC.write(ADC.read(0))

def destroy():
    ADC.write(0)

if __name__ == "__main__":
    try:
        setup()
        loop()
    except KeyboardInterrupt:
        destroy()
```

Now, turn the shaft of the potentiometer on PCF8591, and you can see the luminance of the LED change and a value between 0 and 255 printed on the screen.

## Phenomenon Picture

# 17.PCF8591_AD_Converter