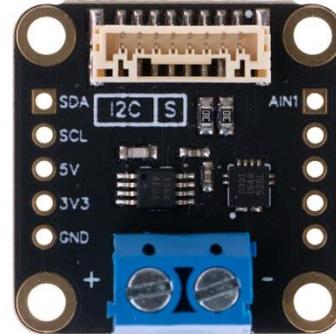


5A Bidirectional Current Sensor – ACS70331 – Trēo™ Module

Module Features

- Allegro ACS70331
- 5mA Precision
- RoHS Compliant
- Software Library
- NightShade Trēo™ Compatible
- Breakout Headers



ACS70331 Features

(from Allegro)

- High-Sensitivity Sensing Up to 5A (AC or DC)
- 1MHz Bandwidth (Response <550ns)
- 1.1mΩ Resistance Results in Low Losses

Applications

- Power Monitoring
- Robotics
- Battery Management
- Solar or Wind Power Stations

Trēo™ Compatibility

Electrical

Communication	I2C
Max Current, 3.3V	7mA
Max Current, 5V	1mA

Mechanical

- 25mm x 25mm Outline
- 20mm x 20mm Hole Pattern
- M2.5 Mounting Holes

Description

The ACS70331 Trēo™ Module is a 5A Bidirectional Current Sensor module that features Allegro’s ACS70331 IC. It can sense ±5A of DC or AC current with a resolution of 5mA. This sensor has an electrical resistance of only 1.1mΩ, resulting in low losses. This module is a part of the NightShade Treo system, patent pending.

Table of Contents

1	Summary	2
2	What is Trēo™?	2
3	Electrical Characteristics	2
4	Electrical Schematic	3
5	Mechanical Outline	4
6	Example Arduino Program	5
7	Library Overview (C++ & Python)	6

1 Summary

The ACS70331 module provides the measurement for a current path. Current flowing from plus to minus is positive. The ACS70331 outputs an analog voltage proportional to the current flow. This voltage is measured with a MAX11644 ADC. The module is initialized with the `begin()` function and then measurements are taken with the `read()` function. The `setOffset()` function can be used to calibrate the sensor zero.

2 What is Trēo™?

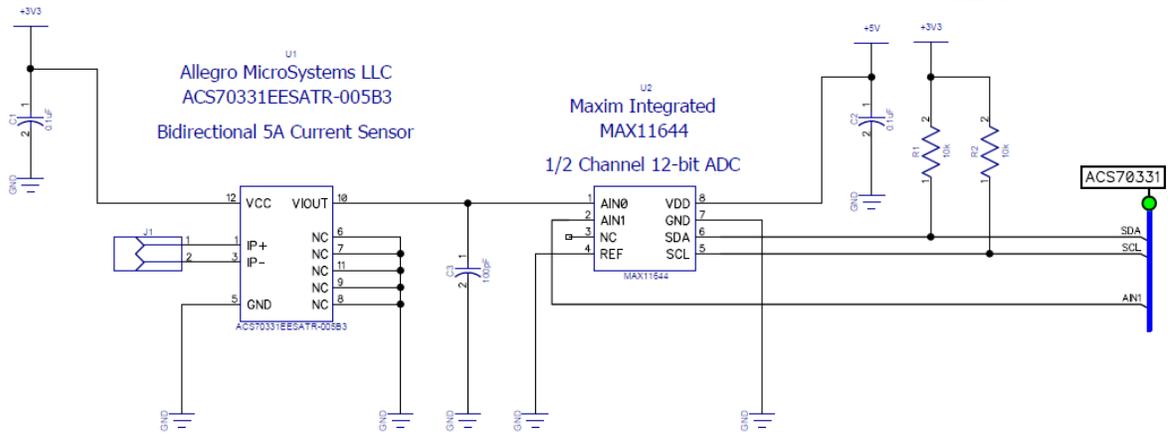
NightShade Trēo is a system of electronic modules that have standardized mechanical, electrical, and software interfaces. It provides you with a way to quickly develop electronic systems around microprocessor development boards. The grid attachment system, common connector/cabling, and extensive cross-platform software library allow you more time to focus on your application. Trēo is supported with detailed documentation and CAD models for each device.

Learn more about Trēo [here](#).

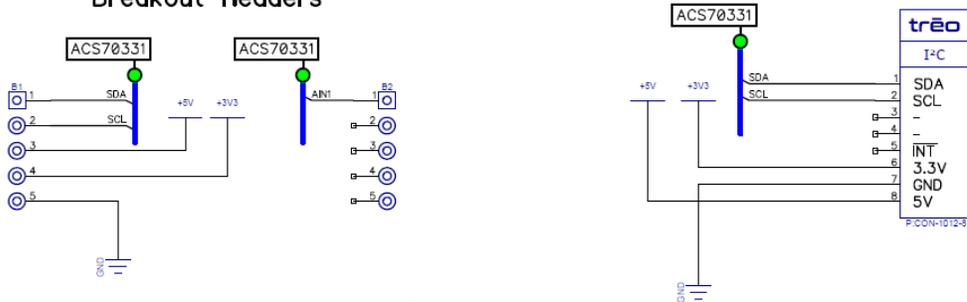
3 Electrical Characteristics

	Minimum	Nominal	Maximum
Voltages			
$V_{i/o}$ (SDA, SCL, INT)	-0.3V	-	3.6V
$V_{3.3V}$	3.1V	3.3V	3.5V
V_{5V}	4.8V	5.0V	5.2V
Measurement			
Bandwidth	-	-	1MHz
Range	-5A AC or DC	-	+5A AC or DC
Precision		5mA/LSB	
Error (25°C)	-5%	-	+5%
Slave Address		0x36	
Operating Temperature	-25°C	-	+85°C

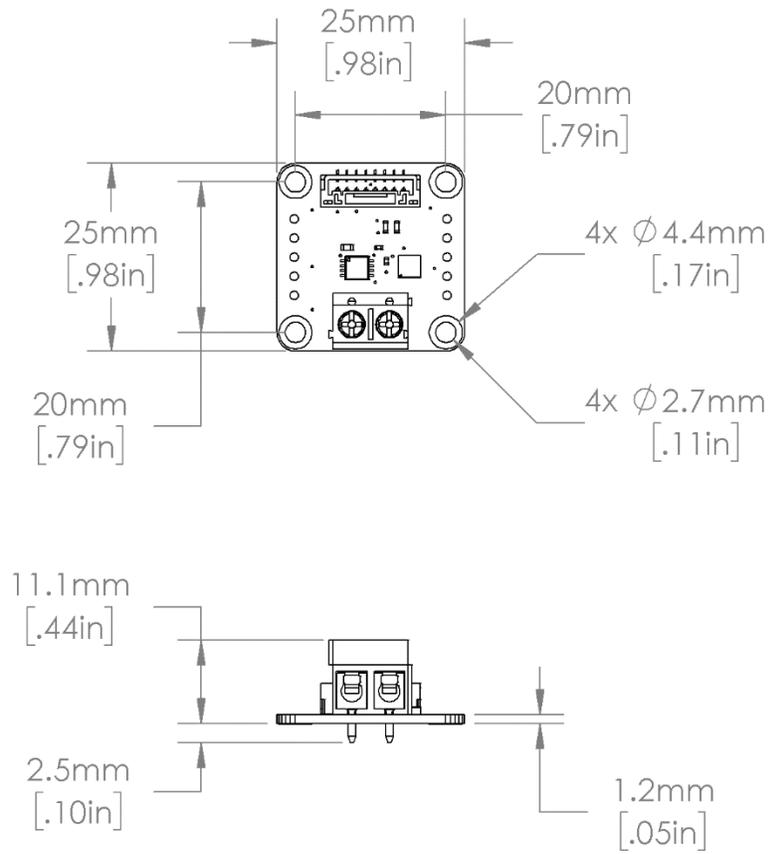
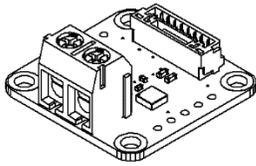
4 Electrical Schematic



Breakout Headers



5 Mechanical Outline



6 Example Arduino Program

```
/*
*****
ACS70331_5A_CurrentSensor - NightShade_Treo by NightShade Electronics

This sketch demonstrates the functionality of the
NightShade Trēo ACS711 5A current sensing module.
(NSE-1130-1) It prints the current value passing
through the sensor to Serial at 115200 baudrate.

Created by Aaron D. Liebold
on February 15, 2021

Links:
NightShade Trēo System: https://nightshade.net/treo
Product Page: https://nightshade.net/product/treo-5a-current-sensor-ac70331/

Distributed under the MIT license
Copyright (C) 2021 NightShade Electronics
https://opensource.org/licenses/MIT
*****/

// Include NightShade Treo Library
#include <NightShade_Treo.h>

// Declare Objects
NightShade_Treo_ACS70331 current(1);

void setup() {
  current.begin();
  Serial.begin(115200);
}

void loop() {
  Serial.print(current.read());
  Serial.println("mA");
  delay(500);
}
```



7 Library Overview (C++ & Python)

C++ Class

```
NightShade_Treo_ACS70331 <classObject>();
```

Python Module

```
<classObject> = NightShade_Treo.ACS70331();
```

7.1 Constructors

NightShade_Treo_ACS70331(int port, uint8_t slaveAddress, uint32_t clockSpeed)

Creates a ACS70331 object.

Arguments:

port	Integer of the I2C port used. (e.g. 0 = "/dev/i2c_0")
slaveAddress	7-bit slave address of the controller
clockSpeed	Desired clock speed for the I2C bus

Returns:

Nothing

NightShade_Treo_ACS70331(int port)

Creates a ACS70331 object assuming the default slave address and clock speed.

Arguments:

port	Integer of the I2C port used. (e.g. 0 = "/dev/i2c_0")
------	---

Returns:

Nothing

7.2 Methods

begin()

Initializes the ACS70331 module.

Arguments:

None

Returns:

Error	0 = Success
-------	-------------



read()

Returns a measurement from the current sensor.

Arguments

None

Returns

Returns the current in mA. (signed int)

setOffset(int offset)

Sets the zero-point offset, for calibration.

Arguments

offset Offset value in mA.

Returns

Nothing