

---

# **Nanoshield\_Termopar Documentation**

***Release 1.0***

**Nanoshield\_Termopar**

**Mar 13, 2017**



---

## Contents

---

<b>1</b>	<b>Class Documentation</b>	<b>3</b>
----------	----------------------------	----------



This is the Arduino software library to access the Termopar Nanoshield, a thermocouple interface module using the MAX31856 IC from Maxim Integrated. It can read several types of thermocouples: K, J, N, R, S, T, E, and B.

- Source code: [https://github.com/circuitar/Nanoshield\\_Termopar](https://github.com/circuitar/Nanoshield_Termopar)
- Documentation: <http://nanoshield-termopar.readthedocs.org/>
- Reference board: [Termopar Nanoshield from Circuitar](#)

Library features include:

- Get external temperature (hot junction)
- Get internal temperature (cold junction)
- Thermocouple type selection
- Continuous conversion mode
- Enable/disable averaging of 2, 4, 8 or 16 samples
- Error checking: open circuit, under/overvoltage and temperature out of range

To install, just click **Download ZIP** and install it using **Sketch > Include Library... > Add .ZIP Library** in the Arduino IDE.

The following **examples** are provided:

- **SerialThermometer**: simple serial port thermometer application.
- **MultiThermometer**: read temperature from multiple Termopar Nanoshields, with different thermocouple types.
- **MultiThermometerCSV**: read temperature from multiple Termopar Nanoshields, generating a CSV output.
- **LcdThermometer**: thermometer application using the [LCD Nanoshield](#) to display the data.
- **RawVoltage**: measure raw thermocouple voltage and cold junction temperature.
- **TemperatureDatalogger**: read temperature from multiple Termopar Nanoshields and log it into an SD card.
- **OpenCircuitDetection**: disable the open circuit detection mode or change its parameters. Might be useful when using long thermocouple wires.



---

## Class Documentation

---

class **Nanoshield\_Termopar**

### Public Functions

**Nanoshield\_Termopar** (uint8\_t *cs* = 8, TcType *type* = TC\_TYPE\_K, TcAveraging *avg* = TC\_AVG\_OFF, TcOcd *ocd* = TC\_OCD\_15\_MS)

Constructor.

Creates an object to access one Termopar Nanoshield.

### Parameters

- *cs*: Chip select pin to access Termopar Nanoshield.
- *type*: Thermocouple type.
- *avg*: Averaging mode.
- *ocd*: Open circuit detection mode.

void **begin** ()

Initializes the module.

Initializes SPI and CS pin.

void **read** ()

Reads all temperatures.

**See** *getInternal()*

**See** *getExternal()*

**See** *hasError()*

double **getExternal** ()

Gets the last external temperature reading (hot junction).

**Return** The last external temperature reading.

**See** [\*read\(\)\*](#)

double **getInternal** ()

Gets the last internal temperature reading (cold junction).

**Return** The last internal temperature reading.

**See** [\*read\(\)\*](#)

bool **isExternalOutOfRange** ()

Checks if external temperature is out of range.

**Return** True if external temperature (hot junction) is out of range.

bool **isInternalOutOfRange** ()

Checks if internal temperature is out of range.

**Return** True if internal temperature (cold junction) is out of range.

bool **isOverUnderVoltage** ()

Checks for overvoltage or undervoltage.

**Return** True if there is overvoltage or undervoltage on the thermocouple inputs.

bool **isOpen** ()

Checks if thermocouple circuit is open.

**Return** True if thermocouple circuit is open.

bool **hasError** ()

Checks if there are errors.

**Return** True if any of the following errors is detected: open circuit, overvoltage, undervoltage, internal temperature out of range or external temperature out of range.

**See** [\*isExternalOutOfRange\(\)\*](#)

**See** [\*isInternalOutOfRange\(\)\*](#)

**See** [\*isOverUnderVoltage\(\)\*](#)

**See** [\*isOpen\(\)\*](#)

---

This documentation was built using [ArduinoDocs](#).



## N

Nanoshield\_Termopar (C++ class), 3  
Nanoshield\_Termopar::begin (C++ function), 3  
Nanoshield\_Termopar::getExternal (C++ function), 3  
Nanoshield\_Termopar::getInternal (C++ function), 4  
Nanoshield\_Termopar::hasError (C++ function), 4  
Nanoshield\_Termopar::isExternalOutOfRange (C++  
function), 4  
Nanoshield\_Termopar::isInternalOutOfRange (C++  
function), 4  
Nanoshield\_Termopar::isOpen (C++ function), 4  
Nanoshield\_Termopar::isOverUnderVoltage (C++ func-  
tion), 4  
Nanoshield\_Termopar::Nanoshield\_Termopar (C++  
function), 3  
Nanoshield\_Termopar::read (C++ function), 3