

QTR-8RC Reflectance Sensor Array

Note: The QTR-8RC reflectance sensor array requires digital I/O lines to take readings. The similar QTR-8A reflectance sensor array is available with analog outputs, and the reflectance sensor is available individually as a QTR-1RC reflectance sensor or QTR-1A reflectance sensor.



Functional Description

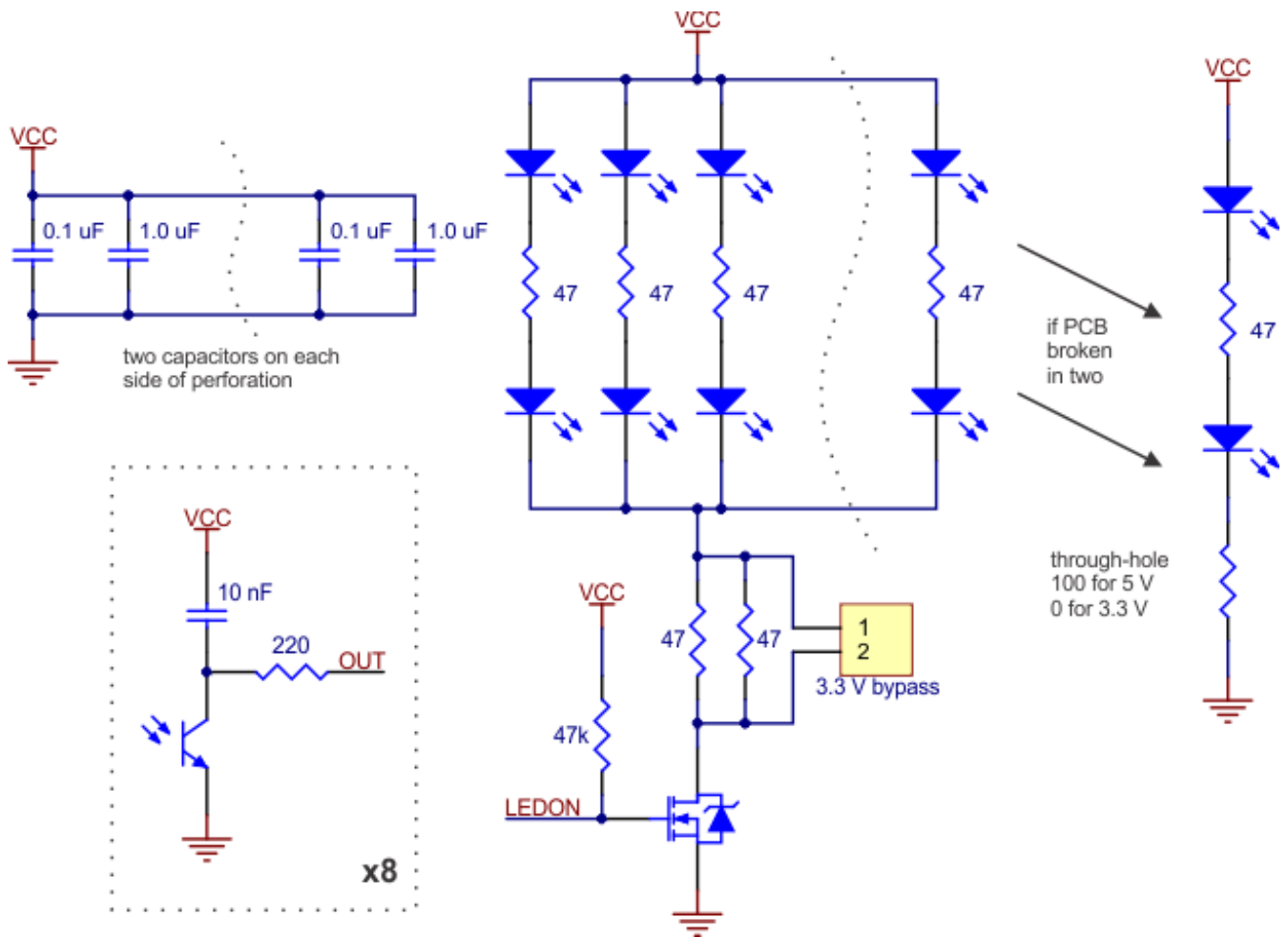
The QTR-8RC reflectance sensor array is intended as a line sensor, but it can be used as a general-purpose proximity or reflectance sensor. The module is a convenient carrier for eight IR emitter and receiver (phototransistor) pairs evenly spaced at intervals of 0.375" (9.525 mm). To use a sensor, you must first charge the output node by applying a voltage to its OUT pin. You can then read the reflectance by withdrawing the externally supplied voltage and timing how long it takes the output voltage to decay due to the integrated phototransistor. Shorter decay time is an indication of greater reflection. This measurement approach has several advantages, especially when coupled with the ability of the QTR-8RC module to turn off LED power:

- **No analog-to-digital converter (ADC) is required**
- **Improved sensitivity over voltage-divider analog output**
- **Parallel reading of multiple sensors is possible with most microcontrollers**
- **Parallel reading allows optimized use of LED power enable option**

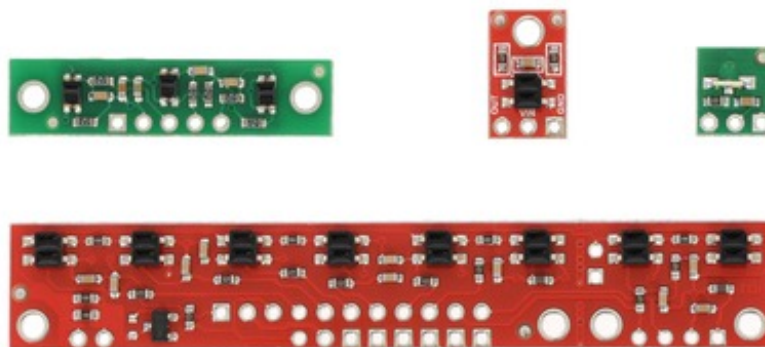
The outputs are all independent, but the LEDs are arranged in pairs to halve current consumption. The LEDs are controlled by a MOSFET with a gate normally pulled high, allowing the LEDs to be turned off by setting the MOSFET gate to a low voltage. Turning the LEDs off might be advantageous for limiting power consumption when the sensors are not in use or for varying the effective brightness of the LEDs through PWM control.

This sensor was designed to be used with the board parallel to the surface being sensed.

The LED current-limiting resistors for 5 V operation are arranged in two stages; this allows a simple bypass of one stage to enable operation at 3.3 V. The LED current is approximately 20–25 mA, making the total board consumption just under 100 mA. The schematic diagram of the module is shown below:



For a similar array with three sensors, consider our QTR-3RC reflectance sensor array. The sensors on the QTR-8RC are also available individually as the QTR-1RC reflectance sensor, and the QTR-L-1RC is an alternative designed to be used with the board perpendicular to the surface.

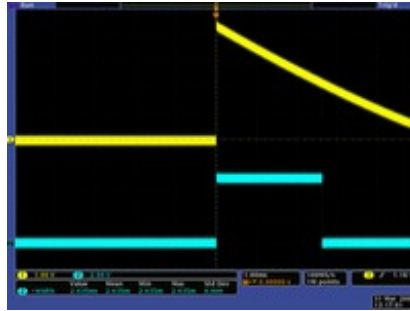


QTR sensor size comparison. Clockwise from top left: QTR-3RC, QTR-1RC, QTR-L-1RC, QTR-8RC.

Specifications

- **Dimensions:** 2.95" x 0.5" x 0.125" (without header pins installed)
- **Operating voltage:** 3.3-5.0 V
- **Supply current:** 100 mA
- **Output format:** 8 digital I/O-compatible signals that can be read as a timed high pulse
- **Optimal sensing distance:** 0.125" (3 mm)

- **Maximum recommended sensing distance: 0.375" (9.5 mm)**
- **Weight without header pins: 0.11 oz (3.09 g)**



QTR-1RC output (yellow) when 1/8" above a black line and microcontroller timing of that output (blue).

Interfacing the QTR-8RC Outputs to Digital I/O Lines

The QTR-8RC module has eight identical sensor outputs that, like the Parallax QTI, require a digital I/O line capable of driving the output line high and then measuring the time for the output voltage to decay. The typical sequence for reading a sensor is:

1. **Turn on IR LEDs (optional).**
2. **Set the I/O line to an output and drive it high.**
3. **Allow at least 10 μ s for the sensor output to rise.**
4. **Make the I/O line an input (high impedance).**
5. **Measure the time for the voltage to decay by waiting for the I/O line to go low.**
6. **Turn off IR LEDs (optional).**

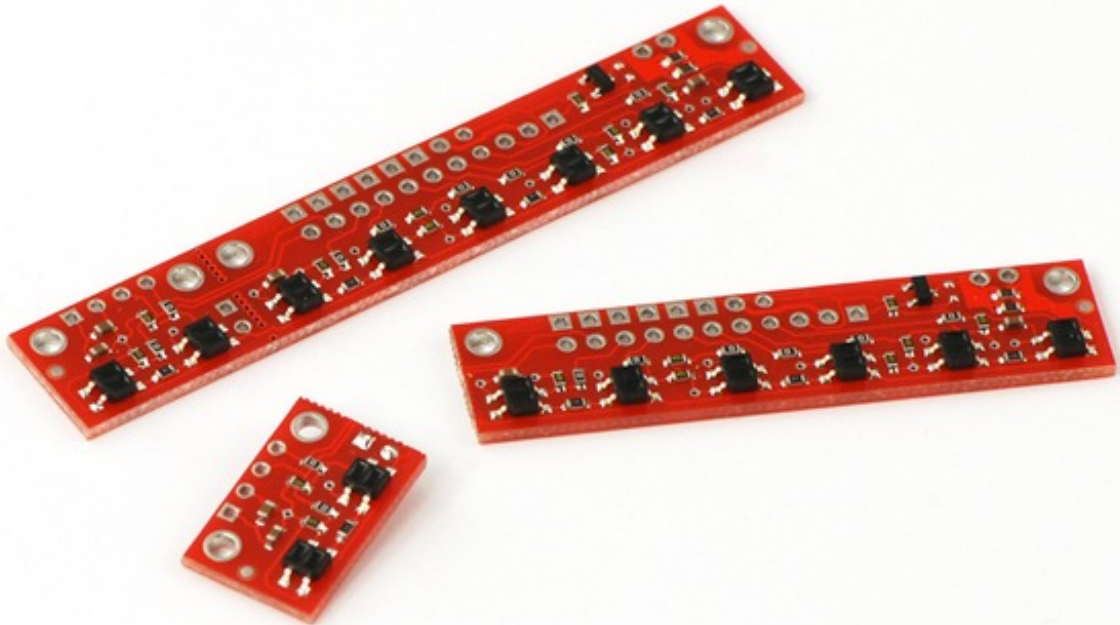
These steps can typically be executed in parallel on multiple I/O lines.

With a strong reflectance, the decay time can be as low as several dozen microseconds; with no reflectance, the decay time can be up to a few milliseconds. The exact time of the decay depends on your microcontroller's I/O line characteristics. Meaningful results can be available within 1 ms in typical cases (i.e. when not trying to measure subtle differences in low-reflectance scenarios), allowing up to 1 kHz sampling of all 8 sensors. If lower-frequency sampling is sufficient, substantial power savings can be realized by turning off the LEDs. For example, if a 100 Hz sampling rate is acceptable, the LEDs can be off 90% of the time, lowering average current consumption from 100 mA to 10 mA.

Our Pololu AVR library provides functions that make it easy to use these sensors with our Orangutan robot controllers; please see the QTR Reflectance Sensors section of our library command reference for more information. We also have a Arduino library for these sensors.

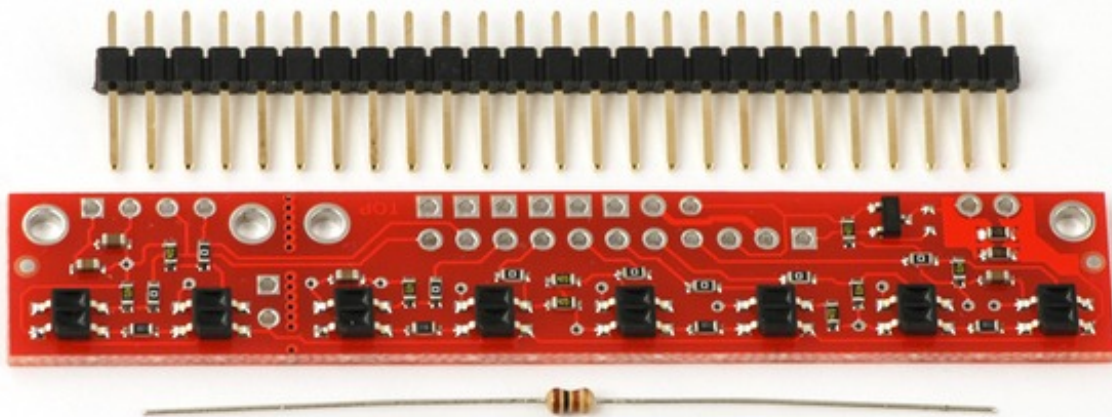
Breaking the Module in Two

If you don't need or cannot fit all eight sensors, you can break off two sensors and still use all 8 sensors as two separate modules, as shown below. The PCB can be scored from both sides along the perforation and then bent until it snaps apart. Each of the two resulting pieces will function as an independent line sensor.



Included Components

This module ships with a 25-pin 0.1" header strip and a 100 Ohm through-hole resistor as shown below.



You can break the header strip into smaller pieces and solder them onto your reflectance sensor array as desired, or you can solder wires directly to the unit or use a right-angle header strip for a more compact installation. The pins on the module are arranged so that they can all be accessed using either an 11×1 strip or an 8×2 strip.

The resistor is required to make the two-sensor array functional after the original eight-sensor array is broken into two pieces. This resistor is only needed once the board has been broken.



Solder the included resistor to the 2-sensor array piece as shown to make the separated piece functional.

How it works in detail

For more information about how this sensor works, see the “How it works in detail” section of the QTR-1RC product page.

[Documentation on producer website.](#)