CANdle
User's Guide

Revision 1.0

Cross The Road Electronics

www.ctr-electronics.com

Table of Contents

TO OUR VALUED CUSTOMERS

It is our intention to provide our valued customers with the best documentation possible to ensure successful use of your CTRE products. To this end, we will continue to improve our publications, examples, and support to better suit your needs.

If you have any questions or comments regarding this document, or any CTRE product, please contact support@crosstheroadelectronics.com

To obtain the most recent version of this document, please visit www.ctr-electronics.com.

1. Device description

The CANdle is a CAN-enabled individually addressable LED controller and a 5V high-efficiency DC voltage regulator all in one compact device.

1.1. Common Use Cases

This device allows users to:

- Produce light using the **eight individually addressable onboard LEDs** - controlled via **CAN bus** (Phoenix) or **WS2812B-compliant pulse-train**. *Good for dome lamp applications.*
- Control **external individually addressable LEDs** via **CAN bus** (Phoenix) **or WS2812B-compliant pulse-train**.
- Power a custom load using a **5V high-efficiency DC voltage regulator** up to **six amperes.**
  Great for Jetson Nano and other popular development boards!
- Measure output current, 5V voltage, and V+ input voltage.
- Enable/disable 5V output programmatically. (1)
- Modulate V+ output for **single-color LEDs**.
- Daisy-chain multiple CANdles using a single source of power.

Note 1: Not available without field-upgrade.

With all these use cases and features, the CANdle will likely find a place in a variety of your LED and voltage control applications.

1.2. Features

- **Eight onboard *individually addressable* LEDs** – CANdle can be used without wiring additional LEDs.
- Supports **multiple kinds** of external **addressable LEDs** (1)
- Works with **12V RGB(W) LED strips** (when $+V_{IN}$ is ~12V)
- Works with **5V RGB(W) LED strips** (using onboard regulated 5V supply)
- 5V Output can also **power 5V peripherals** such as **Raspberry Pi** or **Jetson Nano**
- **Reverse** Input Power **Protection**
- Output **Short Circuit Protection**
- Output **Overcurrent Protection**
- **Thermal Protection**
- Voltage and Current **Measurement**
- High-Side **PWM control** of the **V+ Output**
- **Polycarbonate housing** prevents debris from entering inside device
- **Electrically isolated heatsink** allows for direct mounting to robot frame
- Auto-detection of **CAN bus / CAN FD** (Phoenix) or **WS2812B-compliant pulse-train**.
  No software configuration required to choose between the two.
- **Robust bootloader** and **reliable field-upgrade** (no physical button required, no "stuck states" that requires user intervention)
- Firmware supports **several animations** (Larson, rainbow, twinkle, color fades, etc.) when used with CAN bus
- Robust **embedded wire leads** that will **not shake loose**.
- **Wire leads are labeled** for **fast and easy wiring**.

Note 1: See Section 5 for more information on what types of individually addressable LEDs are supported.

1.3. Kit Contents

The kit contains a single CANdle and two aluminum spacers.

Note: External LED strip is not included, however LED strips may be purchased at ctr-electronics.com.

Note: CANdle has eight onboard individually addressable LEDs. This means CANdle can be illuminated without wiring additional LEDs.

1.4. Electrical Specifications

Symbol	Parameter	Condition	Min	Typ.	Max	Unit
T_{amb}	Ambient temperature		-40		+85	°C
I_Q	Quiescent Supply Current	DC supply 12.0V		50	60	mA
$+V_{IN}$	Supply voltage		6.0	12.0	28.0	V
$-I_{OUT}$	Current sunk thru $-V_{OUT}$		0		6	A
I_{MAX}	Maximum Output Current Sunk via -Vout			6		A
ESD Rating						
	ESD Protection Contact Discharge				±30	kV
	ESD Protection Air-Gap Discharge				±30	kV

1.5. General specifications

Outside Dimensions	2.7" x 1"
Weight (with wires)	1.584 ounces (44.9 grams)
Supported Communication Protocols	CAN 2.0 (1Mbps) CAN FD when used with CANivore Addressable LED Pulse Train
Maximum Operational Temperature @ 5 A(1)	75 °C
Maximum Operational Temperature @ 5 A(2)	55 °C

NOTE 1: This is the maximum expected temperature CANdle can reach when sourcing 5 amps continuously, without any additional heat sink attached
NOTE 2: This is the maximum expected temperature CANdle can reach when sourcing 5 amps continuously, while attached to a heat sink via the provided spacers, shown in Section 4.1 Using the Spacers.

1.6. Onboard LED Specifications

Parameter	Min	Typ.	Max	Unit
Number of onboard LEDs		8		
Red Intensity	700		1000	mcd
Green Intensity	1500		2200	mcd
Blue Intensity	700		1000	mcd
Red Wavelength		625		nm
Green Wavelength		520		nm
Blue Wavelength		470		nm

1.7. LED States

The CANdle features a **tri-color LED** that indicates CAN bus health and general status (similar to other CTR-Electronics CAN devices). This feature can be used to confirm proper CAN bus and power wiring. The table below shows the possible color patterns and their respective CANdle states.

LED Color	5V out / V+ out	Description
Off	Disabled	CANdle is not powered/ plugged in. Check power cabling to the CANdle. Also confirm software configuration has not disabled LED explicitly. **(Note 1)**
Yellow/Green (LED is never off)	Disabled	Device is in bootloader, most likely because field-upgrade failed in middle of event. Inspect CAN bus wiring and re-field-upgrade using Phoenix Tuner. If device has valid firmware, turn device off, wait 10 seconds, and turn device on to boot strap it. In this state, the 5V and V+ outputs are off to prevent accidental shorting.
Red/Yellow (LED is never off)	Disabled (…if firmware determines this is required)	Hardware is damaged.
Slow Red Blink	Configurable/On (Note 2)	Check CAN Bus health and connection to the CANdle/Pixel pulse train connection to the CANdle. In this state, the output of the CANdle is determined by configuration.
Rapid Red Blink	Off	Device has faulted and has disabled outputs to prevent hardware damage. Check Phoenix Tuner self-test to diagnose what fault occurred. Check wiring to ensure no shorts across output leads.
Yellow Blink	Configurable/On (Note 2)	CAN bus detected but no commands are being received. Most likely no CANdle object created in software.
Green Blink	Configurable/On (Note 2)	CANdle is actively controlled either via CAN or through the pixel pulse train.

Note 1: Software configuration allows customers to turn off tri-color LED when loss of signal occurs (CAN bus or pulse-train) as some applications may require absolutely no light when disconnected. In such a case, consider this as a potential cause of why the LED is off.

Note 2: The outputs are controlled by software configuration.

2. Wiring

Due to the sealed nature of the CANdle, all wires leads are pre-installed. Because of this, customers can use whatever connector solution is required for the application.

The wiring details will be determined by the desired use case of the CANdle. As a result this section documents several wiring diagrams for each use-case.

2.1. Wiring Diagram

Color of wire	Signal	Description
Inputs to CANdle		
Red (18 AWG)	$+V_{IN}$	The supply input voltage (6 V – 28 V, typically 12 V)
Black (18 AWG)	$-V_{IN}$	The supply ground (0 V)
Yellow (22 AWG)	CANH (Note 1)	CAN Bus High
Green (22 AWG)	CANL (Note 2)	CAN Bus Low
Outputs from CANdle		
Blue (18 AWG)	$5V_{OUT}$	Regulated 5V for powering 5V addressable LEDs or 5V development boards.
Orange (18 AWG)	$+V_{OUT}$	Provides voltage to LEDs that are rated for supply voltage. This can be modulated via software. A common use case is to power CANdle with 12V lead acid battery, and use $+V_{OUT}$ to power 12V LEDs. **Use $+V_{OUT}$ to power 12V LED strips.** (Note 3) **Do not use $+V_{IN}$ to power 12V LED strips.** (Note 3)
White (18 AWG)	D_{OUT}	WS2812B compliant signal for individually addressable LEDs.
Gray (18 AWG)	$-V_{OUT}$	Ground reference for all outputs. **Do not use $-V_{IN}$ to sink load currents.**

Note 1: Both CANH wires are electrically common.
Note 2: Both CANL wires are electrically common.
Note 3: **CANdle safely disables $+V_{OUT}$ and $5V_{OUT}$ during fault conditions. Therefore, these must be used to power outputs for safe operation.**

2.2. Enclosure Wire labels

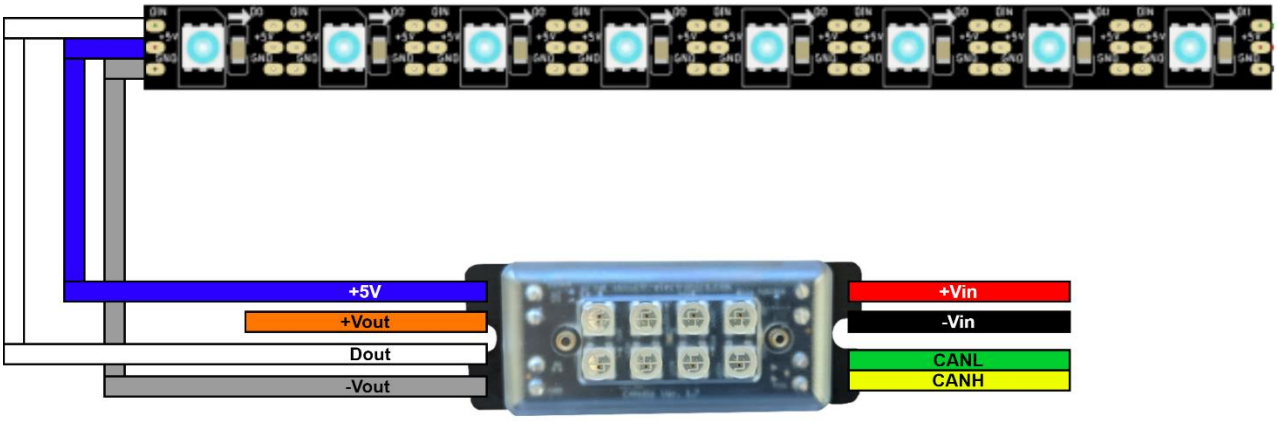
Wire leads labels are available on the back of heatsink. This makes wiring the CANdle even easier.

2.3. Individually Addressable LEDs

The output leads must be used to control individually addressable LEDs.

5V addressable LEDs must use the +**5V rail**.
12V addressable LEDs must use the **+Vout rail**.
Ground return must use -V_{OUT}.
Do not connect +V_{IN} or -V_{IN} to the LED strip**.**

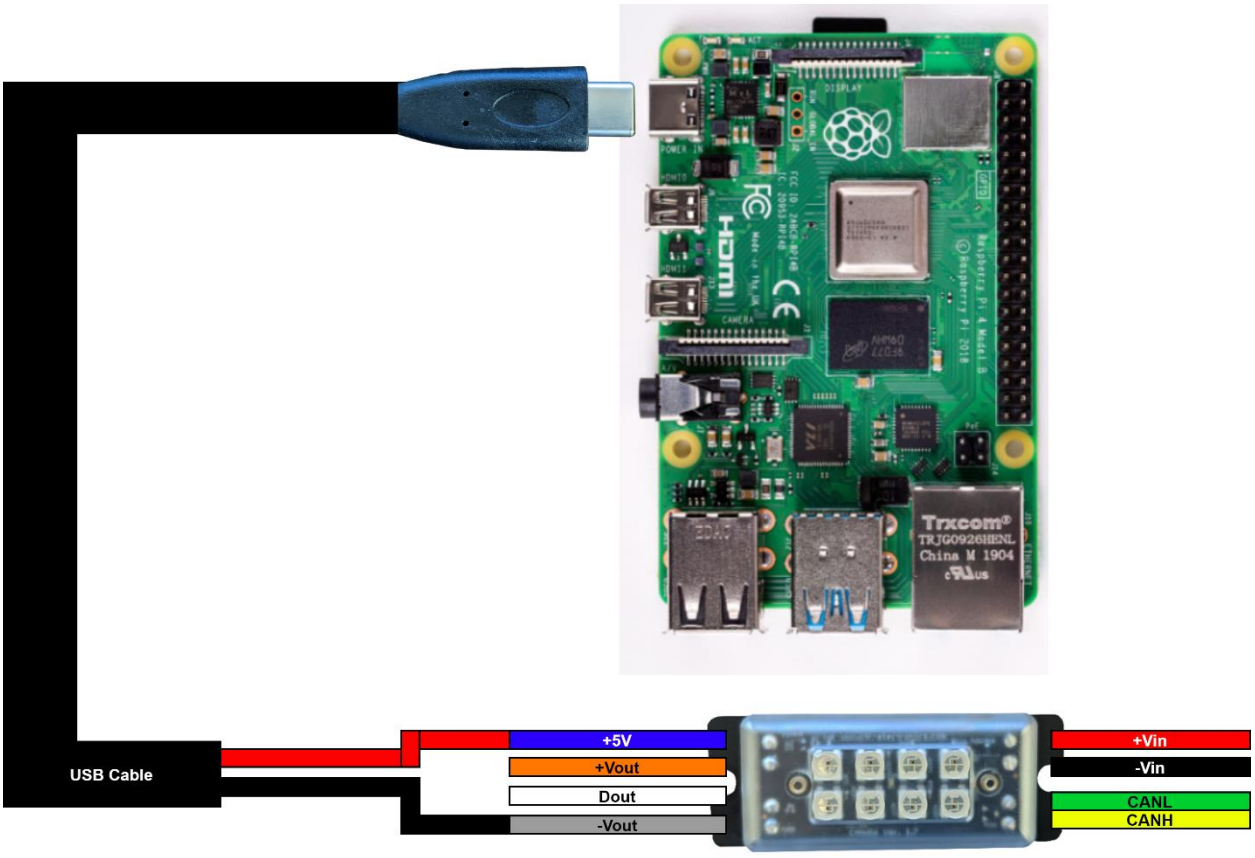
The example below demonstrates wiring an external 5V LED strip.



2.4. Powering a Raspberry Pi

CANdle can be used to power custom 5V devices such development boards.
This allows customers to leverage the power control and monitoring features of CANdle when using a Raspberry Pi subsystem.

Do not connect +V_{IN} or -V_{IN} to the load.

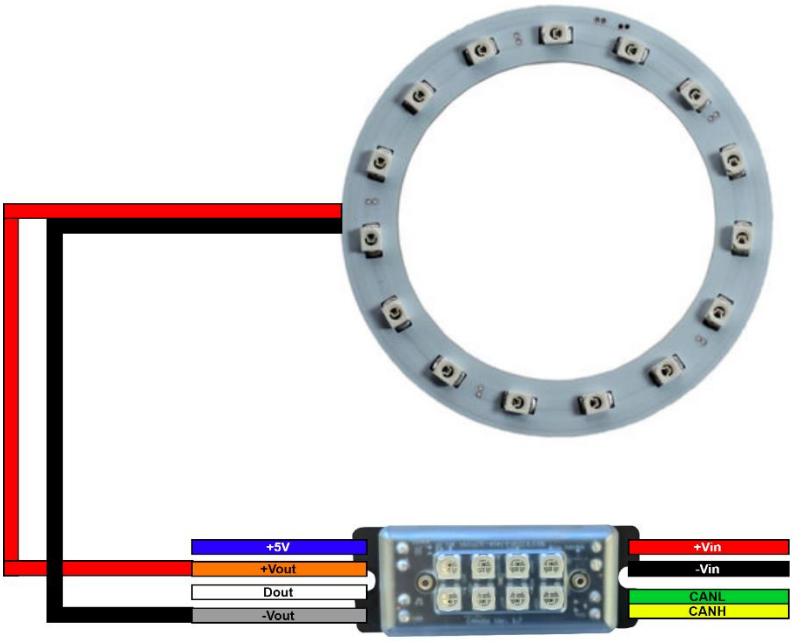


2.5. Controlling a single-color LED

Single-color non-addressable LED strips that simply take direct voltage are also supported with CANdle. These LEDs are not individually addressable – meaning all LEDs will increase with brightness as the output intensifies.
For such LEDs, connect **+V_{OUT}** and **-V_{OUT}** to the LEDs.

Do not connect +V_{IN} or **-V_{IN}** to the LED strip**.**

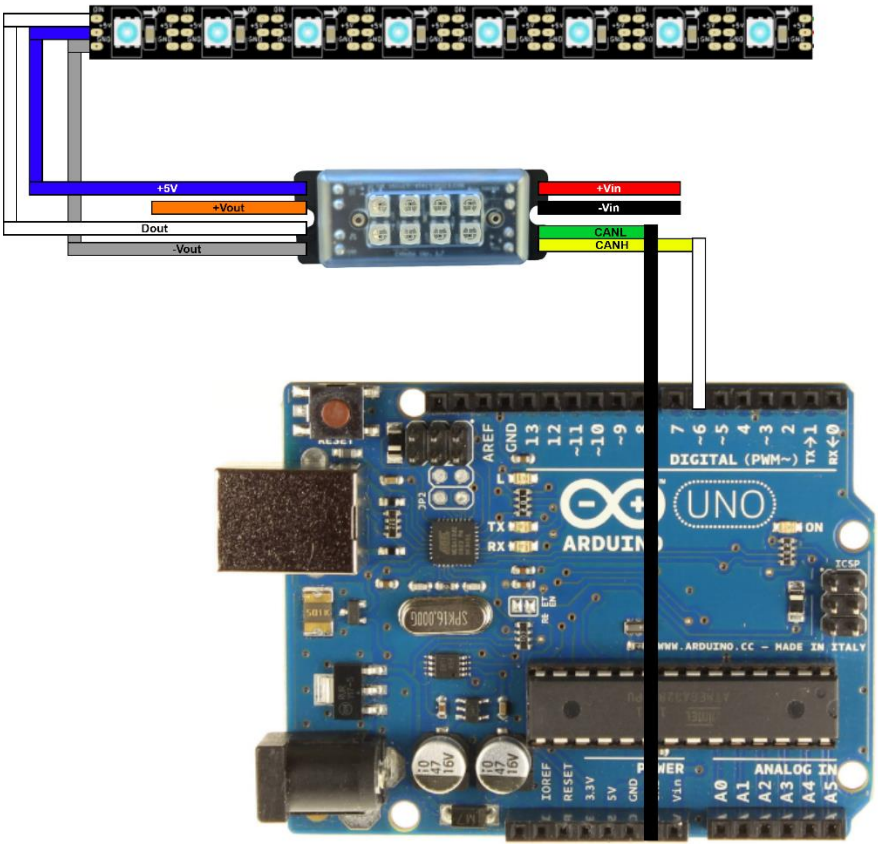
The **LEDs must be rated for** the **same voltage** wired to CANdle via **+V_{IN}**.
If CANdle is wired to ~12V source, the LEDs must be able to withstand the same voltage.



2.6. Pixel Pulse Train Control - Separate Controller

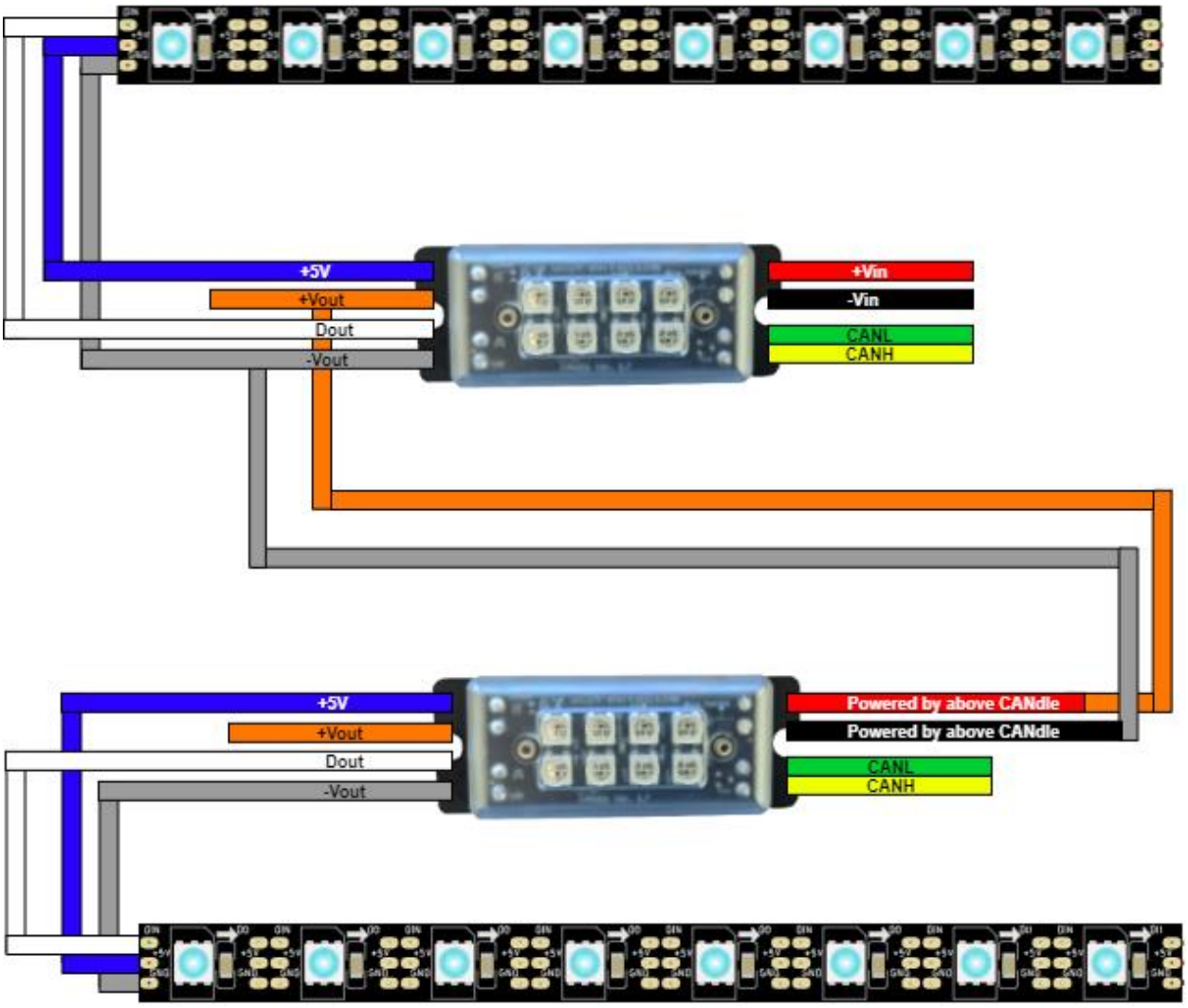
The previous wiring sections demonstrate control and monitoring via the CAN bus wires.
However, CANdle can also **auto-detect a WS2812B-compliant pulse-train** from common generators such as the **Arduino**.

No software configuration is required by CANdle for this, simply wire CANH to the pulse-train signal, and CANL to the logic ground of the signal generator.



2.7. Daisy Chain from Single Power Source

Additionally, a supplemental CANdle can be powered "down-stream" of another CANdle. Note that this will increase the current draw of the "up-stream" CANdle.



3. Control Methods

3.1. Individually Addressable Control

There are two methods for controlling **individually** addressable LEDs. This includes both the eight onboard LEDs and external LEDs (if wired by the user).

Note that in both circumstances, LED index 0 – 7 controls the first eight LEDS. LED index 8 and on reference the external LEDs. See Section 3.2. for more information on LED order.

3.1.1. Individually Addressable Control – CAN bus

CANdle can control individually addressable LEDs by receiving commands from the **CAN bus**. Controlling CANdle in this manner requires use of CTR Electronics' Phoenix API, available for download here: ctr-electronics.com/software.
This allows customers to:

- request pre-written multi-LED animations with a single command
- or direct control each LED manually

3.1.2. Individually Addressable Control - WS2812B compliant pulse train

Individually addressable LEDs can also be controlled by a WS2812B-compliant pulse-train. Connect the pulse train signal to CANH (yellow) and connect the signal generator's ground reference to CANL (green).

The CANdle will still provide power to the downstream LEDs, allowing system developers to only be concerned about color control.

Note that the pixel pulse train still controls the onboard LEDs. The first eight onboard LEDs are first on the data chain, then the ninth LED index will be the first LED on the external LED strip.

For information on the onboard LEDs, see Section 6. On board LEDs.

3.2. On board LEDs and LED Order

The LEDs are sequenced in order as depicted below:

3.3. CAN Single-Color Control

CANdle can also control brightness of single-color LEDs by modulating the supply voltage (via $+V_{OUT}$).

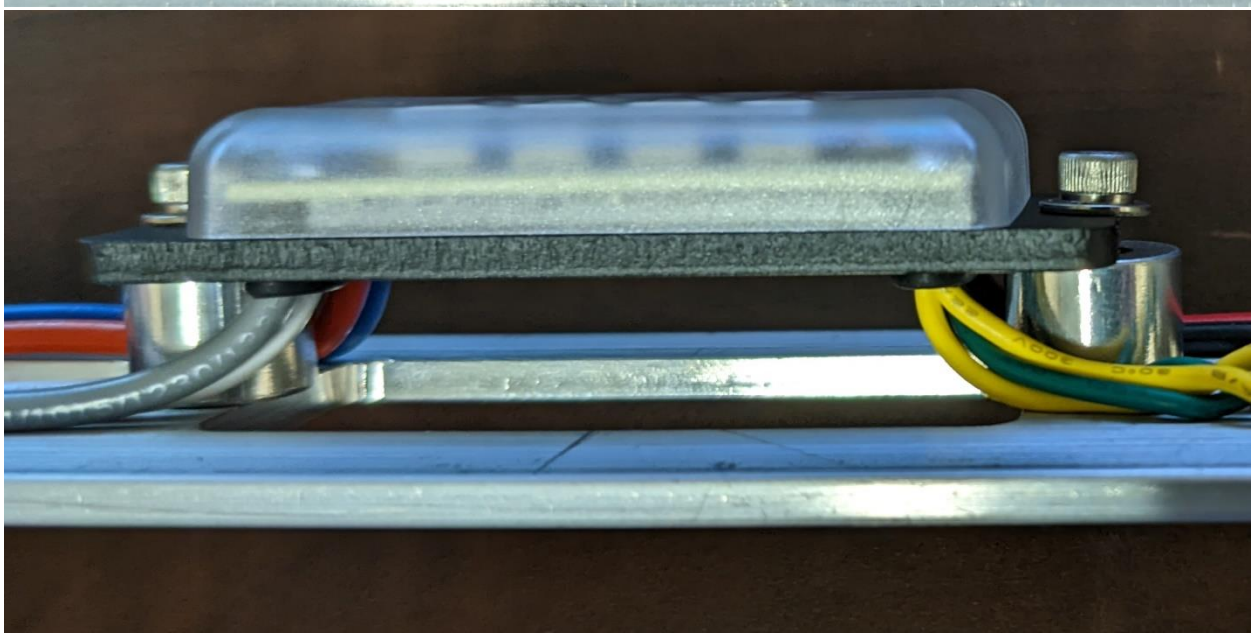
This feature requires using CAN bus and Phoenix API.

4. Mounting

The CANdle allows for multiple methods of mounting, two of which are described below.

4.1. Using the Spacers

The CANdle can be mounted using the provided spacers. This allows clearance for the wires to travel out under the heatsink. This is a particularly convenient method of mounting the CANdle, as it only requires a flat surface to drill holes into and subsequently mount against.



4.2. Panel Mounting

CANdle can also be panel mounted. This requires milling out a hole so that the front face (LED side) can be fitted through. Refer to Section 8 for mechanical dimensions.

5. External LED strip types

There are multiple types of individually addressable LEDs that affect how color is encoded.
When using external LED strips, it is essential to configure the correct type of LED to get the desired effects.

CANdle supports the following three color (24 bit pixel) types:

- GRB – First byte is Green, second byte is Red, third is Blue **(default configuration)**
- RGB – First byte is Red, second byte is Green, third is Blue
- BRG – First byte is Blue, second byte is Red, third is Green

CANdle supports the following four color (32 bit pixel) types:

- GRBW – First byte is Green, second byte is Red, third is Blue, fourth is White
- RGBW – First byte is Red, second byte is Green, third is Blue, fourth is White
- GRBW – First byte is Blue, second byte is Red, third is Green, fourth is White

Failure to properly configure the LED type will result in incorrect colors.

5.1. Testing Recommendations

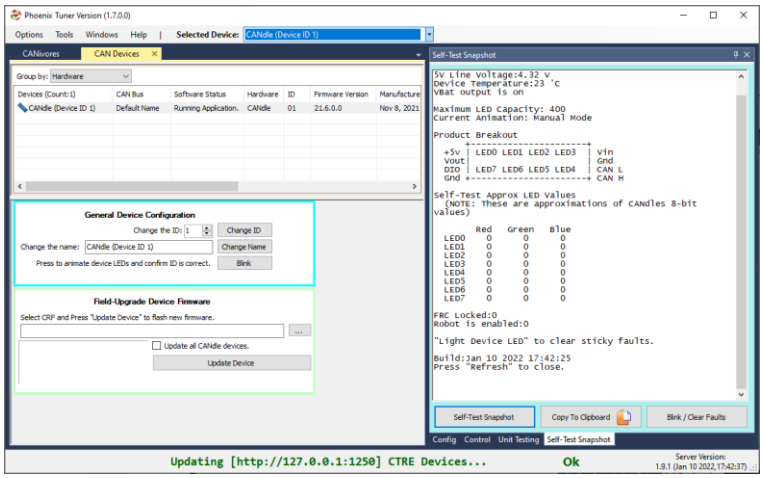
The primary recommendation is to confirm the LED type by **referencing the documentation of the LED strip**.
All LED strips sold by CTR-Electronics have such documentation available on the **product pages** at https://ctr-electronics.com.

But if documentation is not available, the easiest way to configure an unknown LED strip is **to drive a single solid color** (solid red for example). If external LEDs illuminate a **different solid color**, adjust configuration until red appears. Repeat for green and blue.

If requesting a solid color **produces several colors to appear** on the LEDs, then the LED strip likely uses 32 bit pixels (with separate white component) whereas CANdle is configured for three colors (no white component) or vice versa.

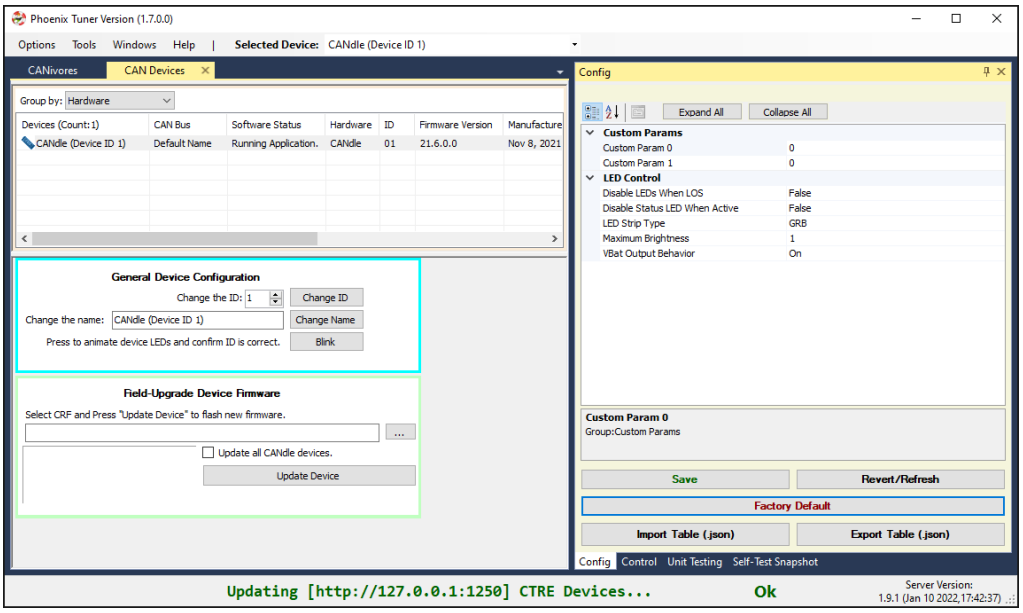
6. Phoenix Tuner / Phoenix Framework

Like all CAN bus devices from CTR-Electronics, CANdle is **supported in Phoenix Framework**.



CANdle is supported in Phoenix Tuner. This allows for fast and efficient debugging of your robot platform

Configuration changes can be done in the graphical interface quickly before incorporating them into your robot application.




```
Robot.java 4
src > main > java > frc > robot > Robot.java > Robot
15   /**
16    * The VM is configured to automatically run this class, and to call the
17    * functions corresponding to
18    * each mode, as described in the TimedRobot documentation. If you change the
19    * name of this class or
20    * the package after creating this project, you must also update the
21    * build.gradle file in the
22    * project.
23    */
24   public class Robot extends TimedRobot {
25
26     CANdle _candle = new CANdle(1, "light bus");
27     WPI_TalonFX _left = new WPI_TalonFX(1, "drivetrain bus");
28     WPI_TalonFX _rght = new WPI_TalonFX(2, "drivetrain bus");
29     Joystick _joystick = new Joystick(0);
30
31
32     @Override
33     public void robotInit() {
34       RainbowAnimation anim = new RainbowAnimation();
35       _candle.animate(anim);
36     }
```

The Phoenix library also supports the CANdle class, allowing for configuration, monitoring, and control of your CANdle from your robot software.

Additional software documentation will be available under the Phoenix Framework documentation section under https://ctr-electronics.com/documentation

7. FAQ

7.1. Is there a way to tell if the device is present/powered?

To determine visually if the sensor is powered and functioning, check the built-in tri-color status LED, see Section 1.7.

7.2. How do I control LEDs?

The CANdle and respective LEDs can be controlled either by the CAN bus or through an external controller generating a pixel pulse train. For more information on the ways to control the CANdle, see Section 3.

Additionally, the choice of how to control the LEDs will determine the wiring as explained in Section 2.

7.3. How many LEDs can I drive from the CANdle?

There are two factors that determine how many LEDs a CANdle can control. Both are covered in sections below.

7.3.1. Software limiting LED count

There is **no *software* limit** when LEDs are controlled **via WS2812B-compliant pulse-train** (through CAN wires).

However, **firmware limits the maximum number of LEDs to 400** when controlled over **CAN bus**.
If this is adjusted in future firmware releases, this can be confirmed in Phoenix Tuner self-test.

7.3.2. Current Draw limiting LED count

The CANdle can supply up to I_{MAX} of total current (combined return current from the $5V_{OUT}$ and $+V_{OUT}$ rail).
Note the return current must be sunk through $-V_{OUT}$.

Typically, this is sufficient for 300 GRB LEDs at max brightness.
Exceeding this may cause LEDs to appear dim or will cause CANdle to fault if current exceeds I_{MAX}.

Limiting the brightness of the CANdle can be used to reduce the total current draw, thus allowing for more LEDs.

7.4. Is the CANdle an LED strip or LED controller?

The CANdle is both an LED strip and an LED controller. The first eight addressable LEDs are onboard and integrated. The first extern LED added via the wire leads is considered the "ninth" addressable LED. This will be references as index '8' in software.

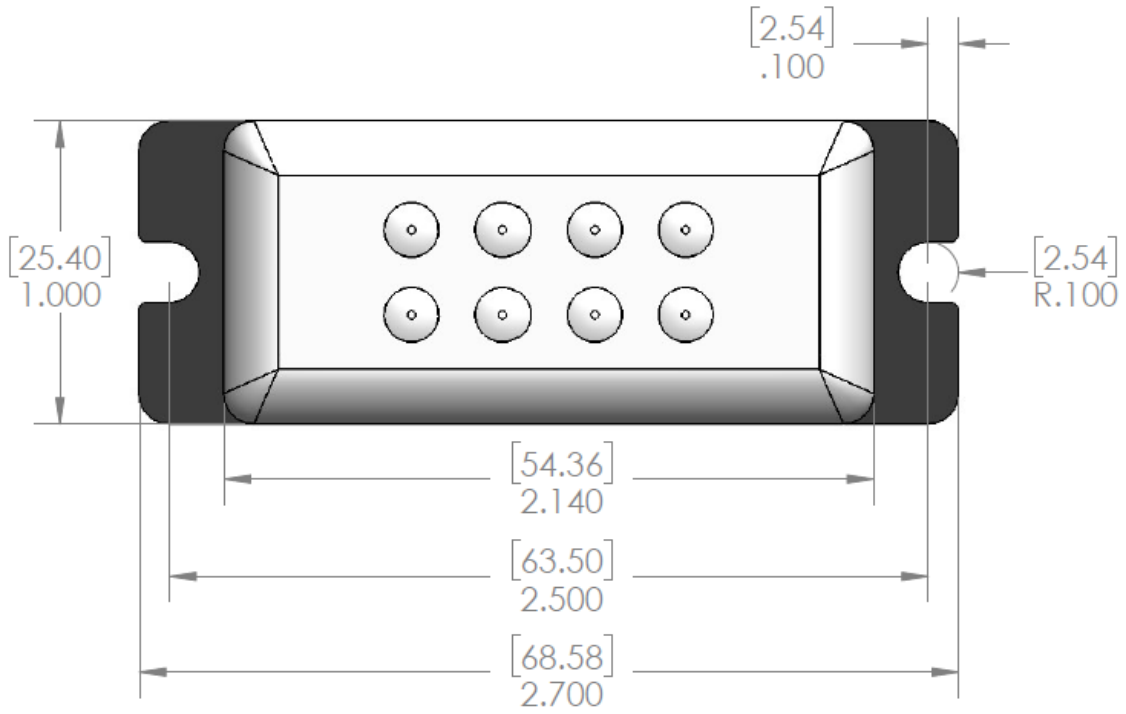
7.5. Do I need to set the RGB type for the onboard eight LEDs?

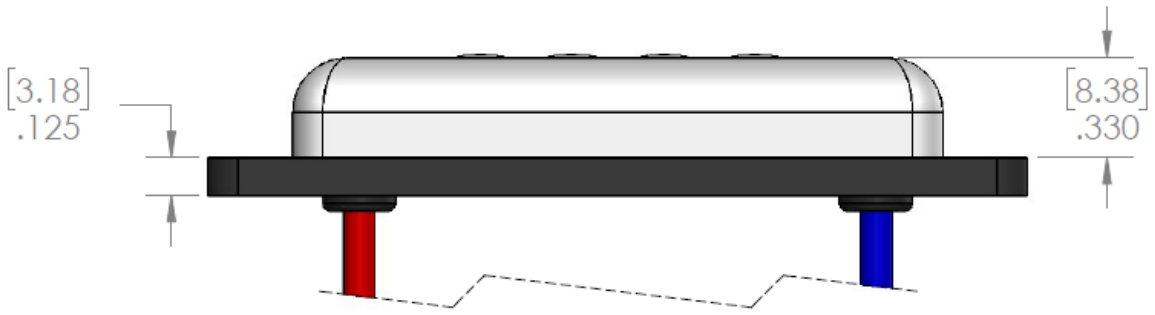
No, the CANdle firmware already *compensates* for *the RGB color type* of the *onboard LEDs*.
You can select **any** external LED strip type that matches the available types in Section 5.

7.6. Can I power my Raspberry Pi or Jetson Nano from the CANdle?

Yes, CANdle can power (non-inductive) loads that rated for 5V and no more than 6amps. Both products in their default configuration require less than 6 amps, and so can be powered by the CANdle. If you are adding peripherals to a co-processor, factor in the current consumption of that co-processor and determine if the total current budget is sufficient for a CANdle to supply. A wiring diagram in Section 2.4 shows how to wire the CANdle to power a Raspberry Pi.

8. Mechanical Drawings





9. Revision History

Revision	Date	Description
1.0	17-Jan-2022	Initial Creation.