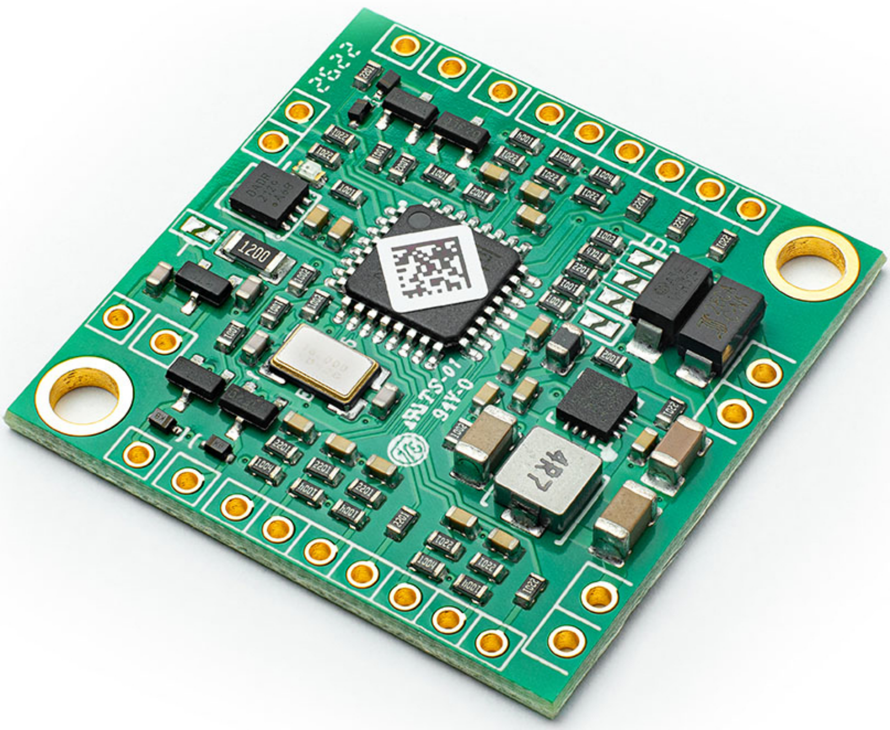
www.ecumaster.com /ecumaster_official /ecumaster

USER MANUAL

CAN Switch Board V3

Document version: 2.1
Firmware version 3.0 or later
Published on: 03 December 2024



ECUMASTER

Contents

1. Device description

The **CAN switch board** is a device that sends information about the state of the following channels via the CAN bus:

- 8 switch inputs (connected to ground)
- 8 analog inputs (voltages from potentiometers, pressure sensors, resistive sensors, etc.)
- 4 low side outputs

The CAN switch board can be used to transmit data from steering wheel buttons, rotary switches, and analog inputs or switches. Additionally, the module serves as a low-side output expansion for the ECU or other CAN bus compatible devices.

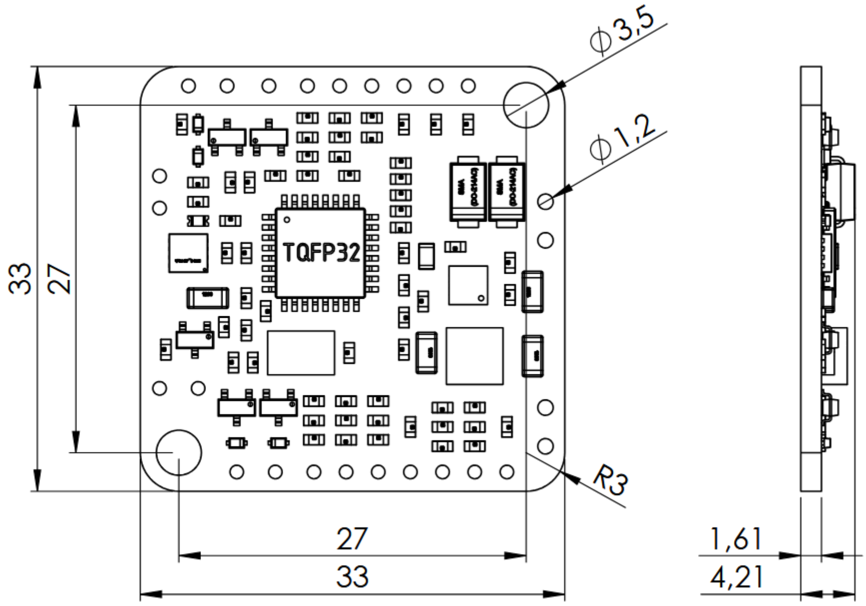
This manual refers to CAN Switch Board version 3, which is backward compatible with version 1 and includes new functionality. Descriptions of backward compatibility and version 1 can be found at the end of the document.

Warning:
This product is not intended for use on public roads.

2. Specification

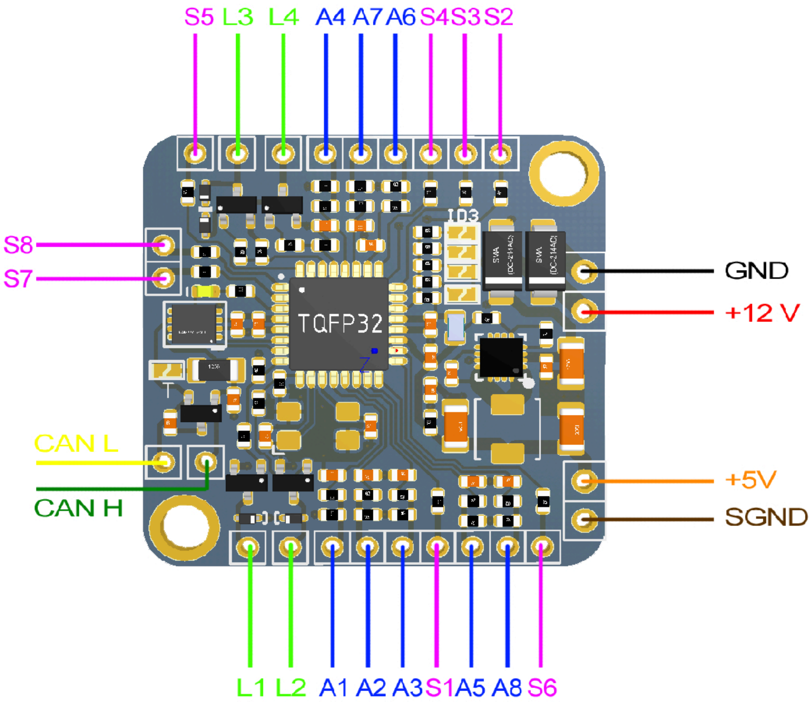
Specification	
Temperature range	AECQ GRADE1 (-40°C to +125°C)
Operating voltage	6-22 V, immunity to transients according to ISO 7637
Analog inputs	8 analog inputs, 0-5 V, 10 bits resolution, 12 V tolerant
Switch inputs	8 switch inputs, switched to ground
Outputs	4 low side outputs, 0.5 A resistive and inductive loads
CAN	CAN 2.0 A/B – 125, 250, 500 (default), 1000 kbps
Weight	5 g
Dimensions	33 x 33 x 5 mm
CAN termination	120 ohm terminator, which is activated by closing a jumper

All dimensions in mm



3. Connection

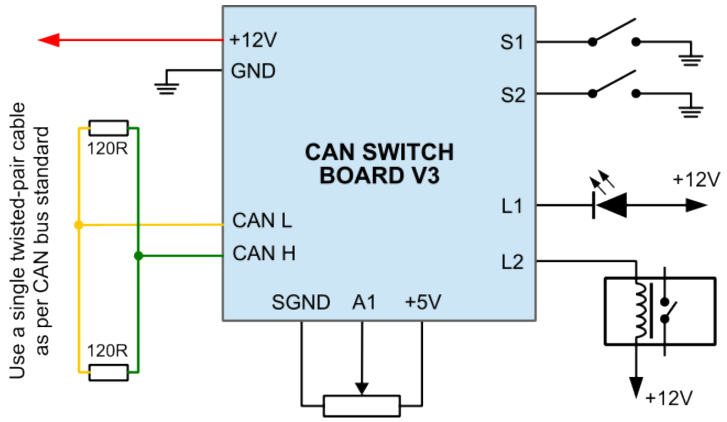
The board requires +12 V switched power (not constant - only when the ignition switch is on). The +5 V supply terminal can be used to power potentiometers or analog sensors. Sensor ground (SGND) should be used for switches, sensors, and potentiometers. All switch inputs are active when connected to ground.



Pin	Description
L1 - L4	Low side outputs, up to 0.5 A each
A1 - A8	Analog inputs, 0-5 V, 10 bits, software-enabled pull-up[1]
S1 - S8	Switch inputs, switched to ground
CAN H/L	CAN bus used for Light Client communication and data output
GND	Sensor supply ground
+12V	Supply voltage for the CAN switchboard
+5V	Sensor supply source
SGND	Sensor supply ground

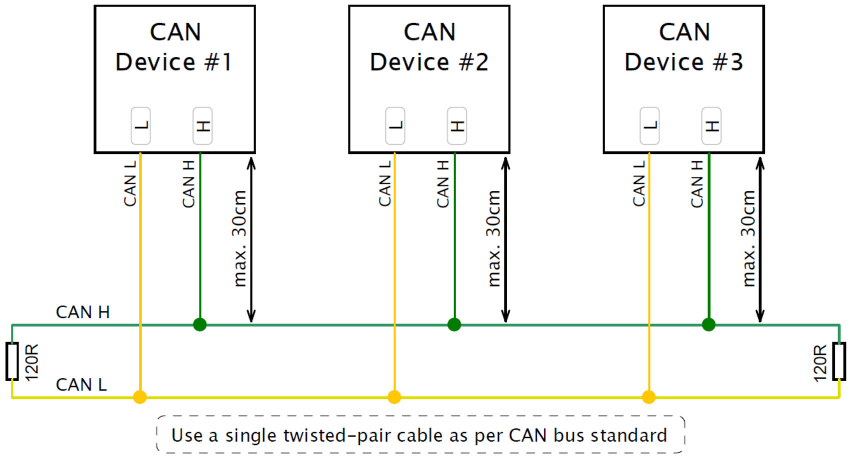
[1]Internal pull-up 35 kOhm ± 15 kOhm, for switches only

Sample connection diagram



The CAN (Control Area Network) bus was developed for communication in automotive applications. Its construction is very simple (only two wires) and its immunity to interference is very high. In modern vehicles, dozens of different electronic modules may communicate via the CAN bus.

Data frames are sent using a network, the topology of which is shown on the following diagram:



In automotive applications, typical data rates on a CAN bus are 1 Mbps, 500 kbps and 250 kbps. The following conditions must be met for each speed:

For a speed of 1 Mbps:

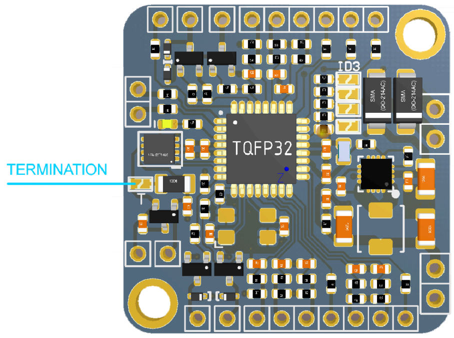
- the maximum length of the connection cable between the bus and the node must not exceed 30 cm
- the maximum bus length is 40 m
- the maximum number of nodes is 30

For a speed of 500 kbps:

- the maximum length of the connection cable between the bus and the node must not exceed 30 cm
- the maximum bus length is 100 m
- the maximum number of nodes is 30

Regardless of the speed, the CAN bus requires termination in the form of 120 ohm resistors at both ends. Additionally, all the connections within the bus must be made using twisted pair wires. It is important that the data transfer speed on a single bus has to be identical for all devices.

It is possible to enable a 120 ohm terminator directly on the CAN Switch Board by closing a jumper.

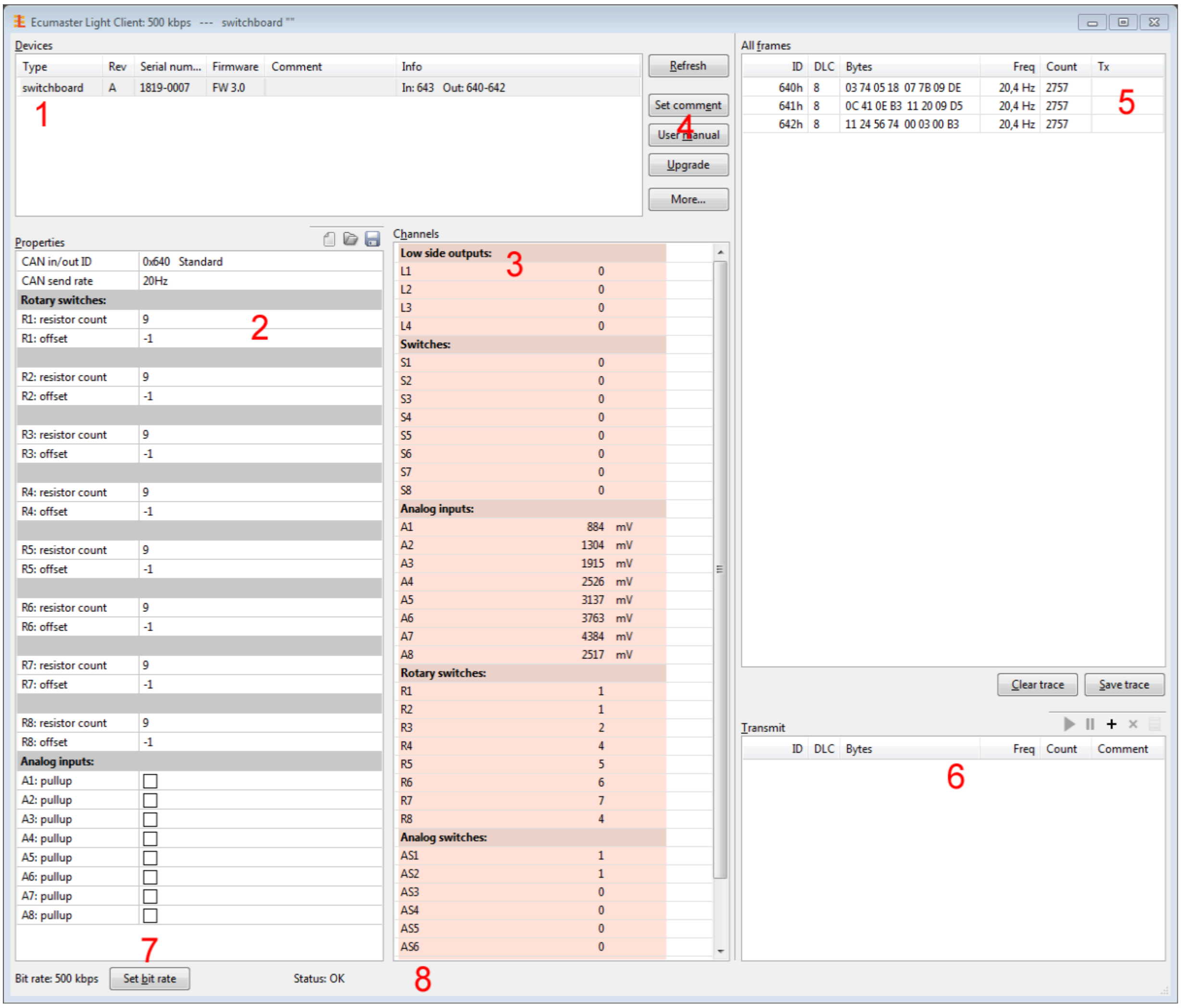


4. Light Client configuration

There are two options for configuring the device: through the Light Client application via the CAN bus or manually using solder joints. The latter option is for users without access to a supported CAN interface (e.g., Ecumaster USB to CAN, Kvaser, Peak PCAN system) or when using version V1. For more information about configuring the device manually, refer to section Configuration V1 *(on page 14)*.

We strongly recommend using the Light Client software to configure both the device parameters and speed.

To use the Light Client for configuration, ensure that jumper ID3 is open (it is open by default). For more information on jumpers, refer to section Configuration V1 *(on page 14)*. Once the application is started and connected to the CAN Switch Board, the following screen will appear.



In the ***Devices*** section of the application (1), there is a list of all *Light Client* compatible devices found on the given CAN bus. Information about each device is displayed, including the name, hardware revision, serial number, firmware version, and additional details about the used CAN

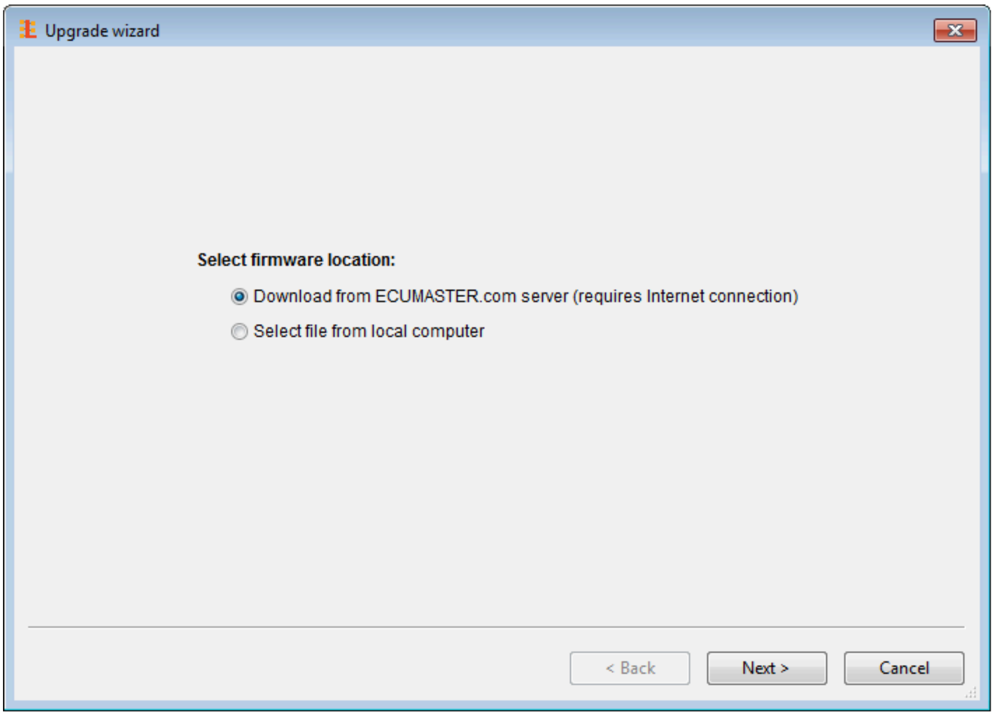
IDs. In this case: switchboard, hardware revision A, serial number 1819-0007, FW 2.1, and CAN ID usage (output: 0x640 to 0x642, and input 0x643). In the ***Properties*** section (2), there is a list of user configured parameters.

Parameter	Description
CAN In/Out ID	The Base ID of the device. CAN Switch Board uses the following IDs: Base ID + 0 to Base ID + 2 as output IDs and Base ID + 3 as an input ID. By default the Base ID is equal to 0x640
CAN send rate	This parameter defines how often the device sends information to the CAN bus
R#: resistors count	This parameter defines the number of the resistors in resistor network used for rotary switch for given analog input. It is used for calculating the position of the rotary switch
R#: offset	The offset for calculated rotary switch position
A#: pullup	Enable the software-selectable pullup resistor (20–50 kΩ range) for the specified analog input. Use this pullup resistor if you intend to use the analog input for a ground-activated switch.

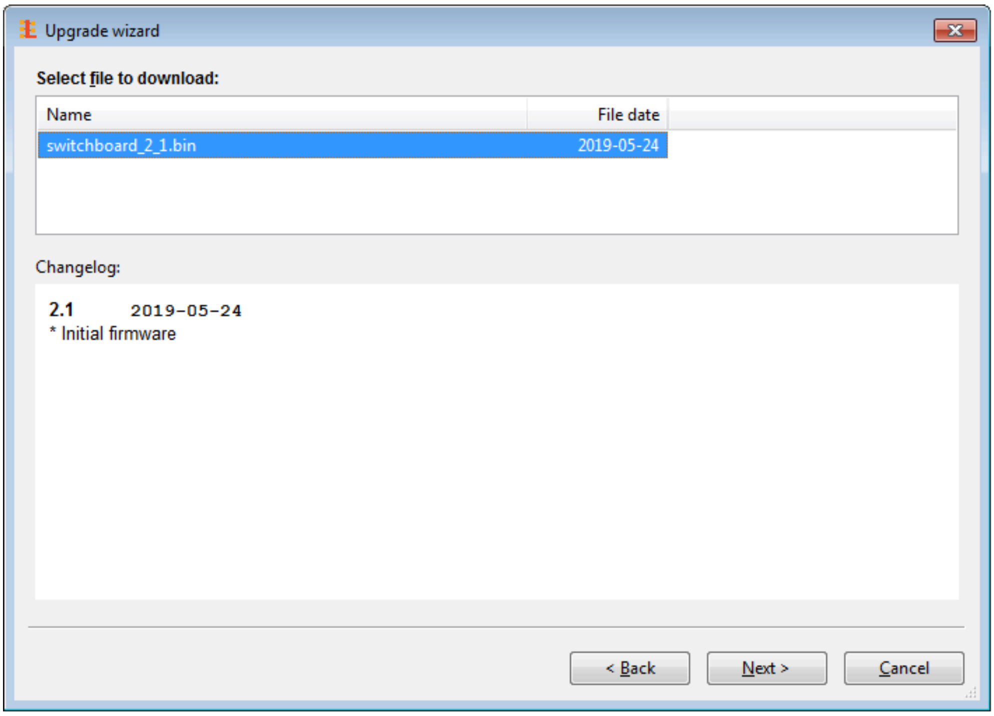
The ***Channels*** section (3) is used to monitor the status of the device's inputs and outputs.

Channel	Description
L1 - L4	The status of low side outputs
S1 - S8	The status of switch inputs
A1 - A8	The voltage (in mV) of analog inputs
R1 - R8	The position of rotary switches connected to the analog inputs
AS1-AS8	The status of analog inputs. Status is equal 1 when the voltage is greater than 3.0 V. and 0 when the voltage is lower than 1.0 V

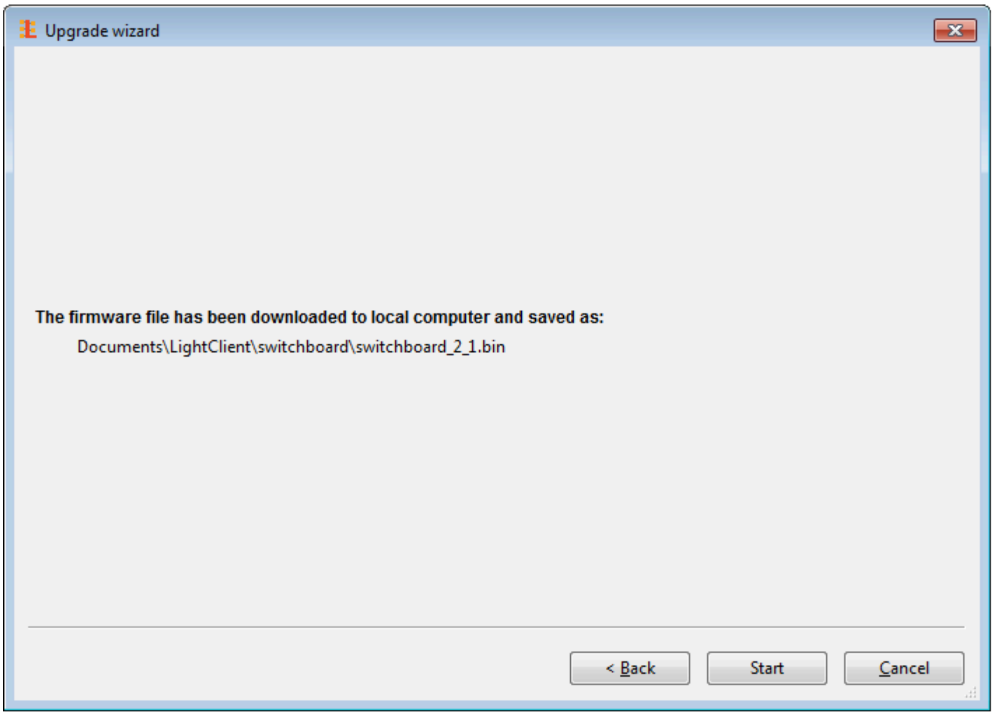
The *Light Client* application allows you to upgrade the connected device’s firmware to the latest version. To do this, press the *Upgrade* button (4). The following screen will appear:



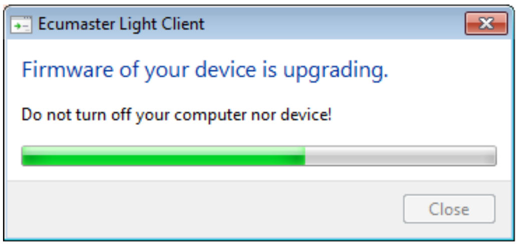
It is strongly recommended that you always download the latest firmware from the Ecumaster server (an internet connection is required). When you press the *Next* button, the following dialog should appear, allowing you to download the selected firmware:



When you press the *Next* button again, the application should display a download confirmation.



Next, press the *Start* button to upgrade the firmware. During the upgrade process, do not turn off your PC or interrupt power to the device!



5. CAN stream

The default base ID is 0x640 and can be changed using Light Client software.

<table>
<tr><th>ID</th><th>DLC</th><th colspan="2">Byte 0</th><th colspan="2">Byte 1</th><th colspan="2">Byte 2</th><th colspan="2">Byte 3</th><th>Byte 4</th><th>Byte 5</th><th>Byte 6</th><th>Byte 7</th></tr>
<tr><td>Base ID</td><td>8</td><td colspan="4">Analog#1(mV)</td><td colspan="4">Analog#2 (mV)</td><td colspan="2">Analog#3 (mV)</td><td colspan="2">Analog#4 (mV)</td></tr>
<tr><td>Base ID+1</td><td>8</td><td colspan="4">Analog#5(mV)</td><td colspan="4">Analog#6(mV)</td><td colspan="2">Analog#7 (mV)</td><td colspan="2">Analog#8 (mV)</td></tr>
<tr><td>Base ID+2</td><td>8</td><td>R1</td><td>R2</td><td>R3</td><td>R4</td><td>R5</td><td>R6</td><td>R7</td><td>R8</td><td>SW_MASK</td><td>AS_MASK</td><td>LS_MASK</td><td>Heartbeat</td></tr>
</table>

Channel	Description
Analog #1 to #8	Voltage value from analog #X input 0-5000 mV, big endian
R1 to R8	Rotary switch position connected to the given analog input. The rotary switch position is represented by 4 bits, two rotary switches are combined in each byte.
SW_MASK	Bit mask representing state of each switch input. The switch 1 is represented by bit 0, the switch 2 is represent by bit 1, and so on
AS_MASK	Mask representing state of analog inputs. If the given analog input value is lower than 2 V the representing bit is 0, else if value is higher than 3 V representing bit is 1
LS_MASK	The bitmask representing the state of the low side outputs. Low side output 1 is represented by bit 0, output 2 by bit 1 and so on
Heartbeat	Counter incremented every sent message

Bit ordering

❑ ❑ ❑ ❑ ❑ ❑ ❑ ❑

7 6 5 4 3 2 1 0

Byte	Bit	Channel	Data Type	Range	Multiplier /Divider	Factor	Offset	Unit
CAN in/out ID+0 (default: 0x640)								
0..1		Analog 1	16-bit U	0 – 65535	1/1	1	0	mV
2..3		Analog 2	16-bit U	0 – 65535	1/1	1	0	mV
4..5		Analog 3	16-bit U	0 – 65535	1/1	1	0	mV
6..7		Analog 4	16-bit U	0 – 65535	1/1	1	0	mV
CAN in/out ID+1 (default: 0x641)								
0..1		Analog 5	16-bit U	0 – 65535	1/1	1	0	mV
2..3		Analog 6	16-bit U	0 – 65535	1/1	1	0	mV
4..5		Analog 7	16-bit U	0 – 65535	1/1	1	0	mV
6..7		Analog 8	16-bit U	0 – 65535	1/1	1	0	mV
CAN in/out ID+2 (default: 0x642)								
0	4..7 (0xF0)	Rotary 1	4-bit U	0 – 15	1/1	1	0	
	0..3 (0x0F)	Rotary 2	4-bit U	0 – 15	1/1	1	0	

Byte	Bit	Channel	Data Type	Range	Multiplier /Divider	Factor	Offset	Unit
1	4..7 (0xF0)	Rotary 3	4-bit U	0 – 15	1/1	1	0	
	0..3 (0x0F)	Rotary 4	4-bit U	0 – 15	1/1	1	0	
2	4..7 (0xF0)	Rotary 5	4-bit U	0 – 15	1/1	1	0	
	0..3 (0x0F)	Rotary 6	4-bit U	0 – 15	1/1	1	0	
3	4..7 (0xF0)	Rotary 7	4-bit U	0 – 15	1/1	1	0	
	0..3 (0x0F)	Rotary 8	4-bit U	0 – 15	1/1	1	0	
4	7 (0x80)	Switch 8	1-bit	0 – 1	1/1	1	0	
	6 (0x40)	Switch 7	1-bit	0 – 1	1/1	1	0	
	5 (0x20)	Switch 6	1-bit	0 – 1	1/1	1	0	
	4 (0x10)	Switch 5	1-bit	0 – 1	1/1	1	0	
	3 (0x08)	Switch 4	1-bit	0 – 1	1/1	1	0	
	2 (0x04)	Switch 3	1-bit	0 – 1	1/1	1	0	
	1 (0x02)	Switch 2	1-bit	0 – 1	1/1	1	0	
	0 (0x01)	Switch 1	1-bit	0 – 1	1/1	1	0	
5	7 (0x80)	Analog State 8	1-bit	0 – 1	1/1	1	0	
	6 (0x40)	Analog State 7	1-bit	0 – 1	1/1	1	0	
	5 (0x20)	Analog State 6	1-bit	0 – 1	1/1	1	0	
	4 (0x10)	Analog State 5	1-bit	0 – 1	1/1	1	0	
	3 (0x08)	Analog State 4	1-bit	0 – 1	1/1	1	0	
	2 (0x04)	Analog State 3	1-bit	0 – 1	1/1	1	0	
	1 (0x02)	Analog State 2	1-bit	0 – 1	1/1	1	0	
	0 (0x01)	Analog State 1	1-bit	0 – 1	1/1	1	0	
6	3 (0x08)	Low side 4	1-bit	0 – 1	1/1	1	0	
	2 (0x04)	Low side 3	1-bit	0 – 1	1/1	1	0	
	1 (0x02)	Low side 2	1-bit	0 – 1	1/1	1	0	
	0 (0x01)	Low side 1	1-bit	0 – 1	1/1	1	0	
7		Heartbeat	8-bit U	0 – 255	1/1	1	0	

The data transmission rate is user defined (the default value is 20 Hz).

6. Low Side Outputs control

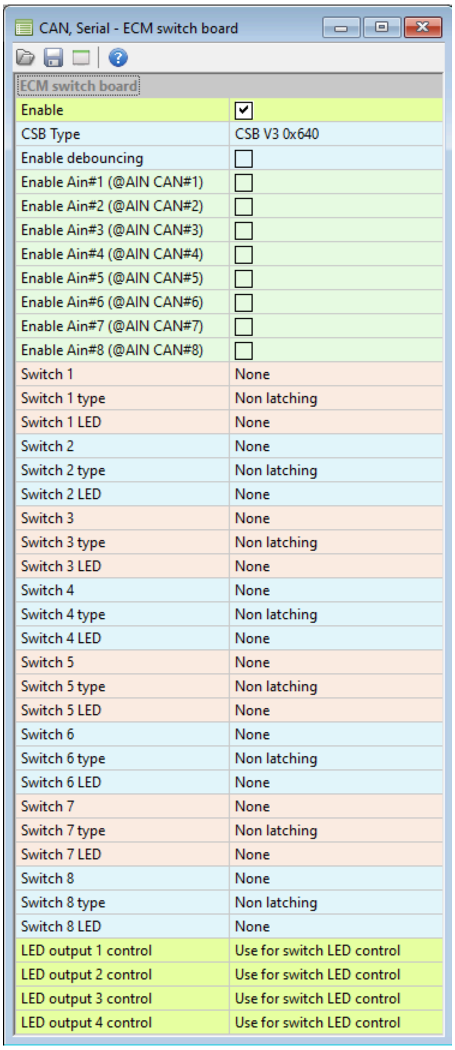
In the V3 data format, the low-side outputs are controlled by the message base ID + 3 (default value is 0x643). This control is available in backward compatibility modes and is described in detail in section Low Side Outputs control V1 *(on page 17)*.

The message format is:

ID	**DLC**	**Byte 0**	**Byte 1**	**Byte 2**	**Byte 3**	**Byte 4**	**Byte 5**	**Byte 6**	**Byte 7**
Base ID + 3	>= 4	**L1 ctrl**	**L2 ctrl**	**L3 ctrl**	**L4 ctrl**	0	0	0	0

7. Using CAN Switch Board V3 with EMU Black

The EMU Black has built in support for CAN Switch Board V3 using a base ID of 0x640. Options for CAN Switch Board can be found in the *CAN, Serial / ECM switch board* menu.

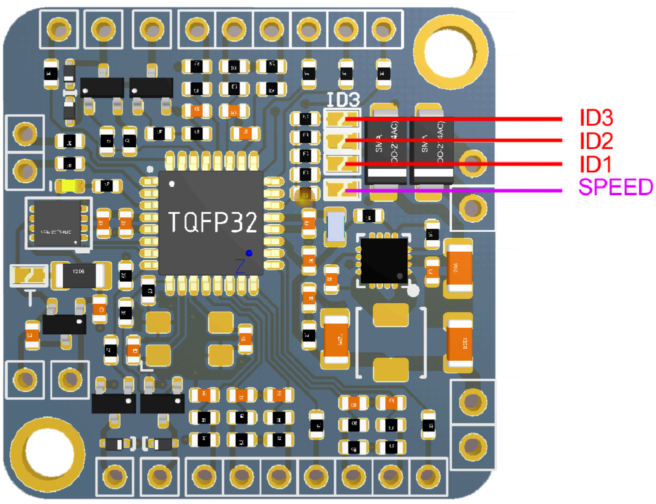


Parameter	Description
Enable	Enables support of CAN Switch Board (CSB)
CSB Type	Allows to select CSB type V1 or V3
Enable debouncing	Enables debouncing for all switches. When debouncing is on, a button must be held for a set time to register a "1" state, filtering out unwanted signals from button contact noise. The set time depends on CAN speed
Enable Ain#X	Overwrites the CAN analog input with the value from the CAN Switch Board
Switch X	Assigns CSB switch input to EMU Black CAN switch
Switch X type	Type of the switch (latching, non latching, multistate)
Switch X LED	Allows to assign CSB low side output that will be active when the switch is pressed
LED output X control	Allows to control CSB low side outputs with the EMU Black parametric outputs

8. CAN Switch Board V1

8.1. Configuration V1

In cases where backward compatibility with version V1 is required, or for users without access to a supported CAN interface, the device should be configured manually using solder joints. The image below shows a list of the available jumpers.



Jumper ID3 is responsible for entering compatibility mode. If it is open, the CAN Switch Board can be configurable using Light Client software (speed, base ID, rotary switch configuration), and the device transmits data using the new V3 data format. The default speed is 500 kbps, and the base ID is 0x640.

When the ID3 jumper is closed with a solder joint, the device enters backward compatibility mode and can be configured using jumpers.

ID1	ID2	ID3	Description
Any	Any	Open	Software configuration by Light Client
Open	Open	Closed	Data Format 0 (compatible with CAN Switch Board V1)
Closed	Open	Closed	Data Format 1 (compatible with CAN Switch Board V1)
Open	Closed	Closed	Data Format 2 (compatible with CAN Switch Board V1)
Closed	Closed	Closed	Data Format 3 (compatible with CAN Switch Board V1)

The SPEED jumper is responsible for manually selecting the device speed. If both the ID3 jumper and the SPEED jumper are closed, the device can still be configured using the Light Client, but a firmware upgrade will not be possible.

ID3	Speed	Description
Open	Open	Speed defined by Light Client. Default 500 kbps
Open	Closed	1 Mbps fixed, no firmware upgrade possible
Closed	Open	500 kbps fixed
Closed	Closed	1Mbps fixed

8.2. CAN stream V1

Data transmission rate is 20 Hz

Data format 0

ID	DLC	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
0x334	8	Analog#1(mV)		Analog#2 (mV)		CALPOT 1	CAL POT 2	Switch mask	Heartbeat
ID	**DLC**	**Byte 0**	**Byte 1**	**Byte 2**	**Byte 3**	**Byte 4**	**Byte 5**	**Byte 6**	**Byte 7**
0x335	8	CALPOT 1	SW#1	SW#2	SW#3	SW#4	SW#5	SW#6	SW#7

ID	DLC	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
0x336	8	CALPOT 2	SW#8						

Channel	Description
Analog#1	Voltage value from analog #1 input 0-5000 mV, big endian
Analog#2	Voltage value from analog #2 input 0-5000 mV, big endian
Switch mask	Bit mask of pressed switches (1 means pressed)
CAL POT #1	The discrete value of analog #1 input. The voltage for each value is multiplication of 384 mV
CAL POT #2	The discrete value of analog #2 input. The voltage for each value is multiplication of 384 mV
Heartbeat	Counter incremented every sent message

Data Format 1

ID	DLC	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
0x334	8	Analog#1(mV)		Analog#2 (mV)		CALPOT 1	CAL POT 2	Switch mask	Heartbeat

Channel	Description
Analog#1	Voltage value from analog #1 input 0-5000 mV, big endian
Analog#2	Voltage value from analog #2 input 0-5000 mV, big endian
Switch mask	bit mask of pressed switches (1 means pressed)
CAL POT #1	The discrete value of analog #1 input. The voltage for each value is multiplication of 384 mV
CAL POT #2	The discrete value of analog #2 input. The voltage for each value is multiplication of 384 mV
Heartbeat	Counter incremented every sent message

Data Format 2

ID	DLC	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
0x666	8	Analog#1(mV)		Analog#2 (mV)		CALPOT 1	CAL POT 2	Switch mask	Heartbeat

Channel	Description
Analog#1	Voltage value from analog #1 input 0-5000 mV, big endian
Analog#2	Voltage value from analog #2 input 0-5000 mV, big endian
Switch mask	bit mask of pressed switches (1 means pressed)
CAL POT #1	The discrete value of analog #1 input. The voltage for each value is multiplication of 384 mV
CAL POT #2	The discrete value of analog #2 input. The voltage for each value is multiplication of 384 mV
Heartbeat	Counter incremented every sent message

Data Format 3

ID	DLC	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
0xA1BA2f1	8	Analog#1(mV)		Analog#2 (mV)		CALPOT 1	CAL POT 2	Switch mask	Heartbeat

Channel	Description
Analog#1	Voltage value from analog #1 input 0-5000 mV, big endian
Analog#2	Voltage value from analog #2 input 0-5000 mV, big endian
Switch mask	bit mask of pressed switches (1 means pressed)
CAL POT #1	The discrete value of analog #1 input. The voltage for each value is multiplication of 384 mV
CAL POT #2	The discrete value of analog #2 input. The voltage for each value is multiplication of 384 mV
Heartbeat	Counter incremented every sent message

8.3. Low Side Outputs control V1

When the backward compatibility mode is used, the control of low-side outputs is as follow:

The message ID is:

Base ID	LED control ID
0x334	0x434

Base ID	LED control ID
0x666	0x766
0xA1BA2f1	0xa1ba2f2

The message format is:

ID	DLC	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
	>= 4	**L1 ctrl**	**L2 ctrl**	**L1 state**	**L2 state**	0	0	0	0

To control a particular low-side output (L output), the corresponding **L control flag** must be set to **1**, which sets the state of that output. For example:

To turn on L1: 0x766, 8, 1,0,1,0 must be sent
To turn off L2: 0x766, 8, 0,1,0,0 must be sent
To turn on L1 and L2 : 0x766, 8, 1,1,1,1 must be sent

9. Document history

Version	Date	Changes
1.0	2019.07.12	Initial revision
1.1	2019.07.31	ID2 must be Open for Data Format 0 (compatible with V1) Changed description of LS_MASK
1.2	2020.02.10	Changed the description of “A#: pullup” resistors for analog inputs
1.3	2020.02.14	Formatting of the document has been corrected
2.0	2024.11.05	New Ecumaster standard layout applied The structure and text have been refined and improved for better readability and clarity
2.1	2024.12.02	Clarified specifications for the internal pullup resistor