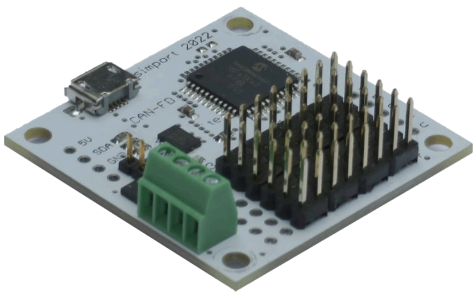
CAN in Simulation

Analog Module



Rev. 1.2

Analog Module

The Analog module is specifically designed to accommodate a maximum of eight analog inputs and eight digital inputs. Each input undergoes a 12-bit conversion process and is carefully filtered to prevent excessive noise interference. The conversion itself is ratiometric to the 5V supply provided by the board, allowing for input usage within the 0 to 5V range. Simple potentiometers can be effortlessly utilized by establishing connections to both the 5V and ground terminals.

As for the digital inputs, they employ incremental encoders capable of counting up or down 12-bit values in slow or fast increments, based on the direction and speed of rotation.

CAN ID	node ID	data type	service code	message code	data byte 0	data byte 1	data byte 2	data byte 3
718h	node	0Dh	item	num	data1	data0	0	0

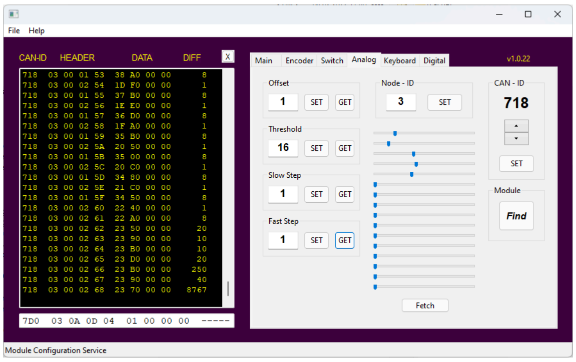
CAN Message

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
data1	b11	b10	b9	b8	b7	b6	b5	b4
data0	b3	b2	b1	b0	0	0	0	0

Data Bytes

The data byte is shifted 4 bits to the left (equivalent to multiplying by 16), resulting in a 16-bit value.

This is how the *Configuration Tool* views an Analog Module (more on page 6):



The CAN-ID is displayed as **718** (hexadecimal), and the node ID is identified as **3**.

The Find button searches for any attached Analog Modules, which is useful when the modules are changed.

There is four parameter available that can be altered upon request:

Offset The Encoder module is capable of handling up to 8 incremental encoders, each of which is assigned a unique ID.

Starting with the *offset* value, the 8 encoders are given consecutive ID values, which will be included in the CAN message sent by the board.

Since the ID values are 1 byte wide, up to 256 different encoders can be distinguished under a given Node-ID.

Threshold The *threshold* is a time value, measured in milliseconds, that serves as a threshold for distinguishing between a fast pulse train and a slow one.

The speed of the encoder turning is determined by the time between two consecutive encoder events (clicks). If the time between these events is below the threshold value, the *fast* flag in the CAN data byte is set, indicating that the encoder is turning quickly. Conversely, if the time between events is above the threshold, the *fast* flag is cleared.

Slow Step If an encoder is turned *slowly*, meaning the time between consecutive events (or clicks) is above the *threshold* value, then the analog value is incremented or decremented by the *slow step* with each event.

Fast Step If an encoder is turned *quickly*, meaning the time between consecutive events (or clicks) is below the *threshold* value, then the analog value is incremented or decremented by a the *fast step* with each event.

Parameter Setting

To modify the parameters of a module, the Module Configuration Service (MCS) is utilized. The MCS is assigned a unique CAN-ID of `7D0h` (equivalent to decimal value `2000`):

CAN ID	node ID	data type	service code	message code	data byte 0	data byte 1	data byte 2	data byte 3
`7D0h`	`node`	`0Ah`	`0Dh`	`pid`	`data`	`0`	`0`	`0`

node ID:	CAN node ID (*node*)
data type:	UCHAR (*`0Ah, 10d`*)
service code:	MCS (*`0Dh`*)
message code:	Parameter index (*`pid`*)
message data:	Parameter value (*`data byte 0`*)

The parameter ID (pid) is used to identify which specific parameter needs to be modified. Data byte 0 contains the value of the parameter.

index	parameter	value(s)
1	offset	1 … 255
2	threshold	1 … 255 [ms]
3	Slow step	1 … 10
4	Fast step	1 … 16

Upon completion of the parameter modification request, the response message will have a message code of **`0`** (zero) if the operation was successful. However, if the requested parameter is out of the valid range or the parameter ID is invalid, the response message will contain a message code of **`-6`**.

CAN-ID Setting

The CAN-ID range for Analog Board messages is **`718h..71Fh`** (decimal **`1816..1823`**).

To change the CAN-ID of the Analog Board, the CAN Identifier Setting Service (CSS) can be used. The message code should be set to 0.

CAN ID	node ID	data type	service code	message code	data byte 0	data byte 1	data byte 2	data byte 3
7D0h	node	0Ch	0Eh	0	0	0	xh	xl

node ID:	CAN node ID (*node*)
data type:	SHORT2 (*0Ch, 12d*)
service code:	CSS (*0Eh, 14d*)
message code:	0
message data:	New CAN ID high byte (xh, data byte 2) New CAN ID low byte (xl, data byte 3)

Upon completion of the CAN Identifier Setting request, the response message will have a message code of **`0`** (zero) if the operation was successful, or **`-6`** if the ID is out of the valid range.

Node-ID Setting

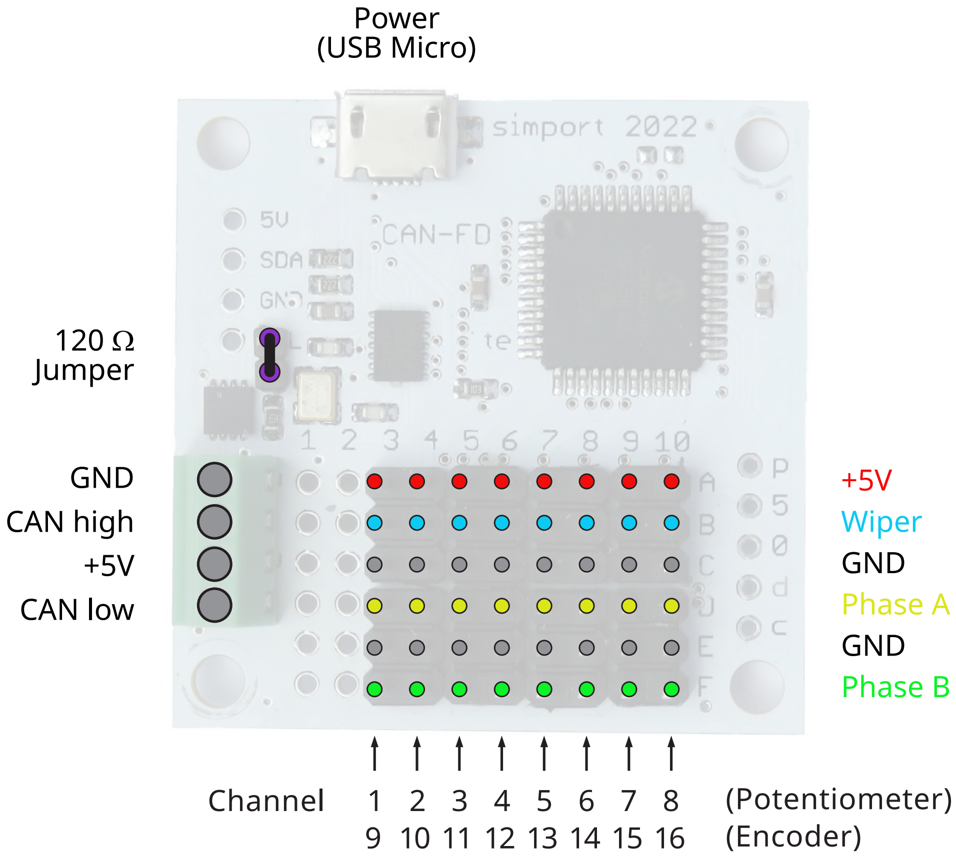
To change the Node-ID of the Analog Board, the Node ID Setting Service (NIS) can be used. Node-ID values are in the range of 1 to 255.

CAN ID	node ID	data type	service code	message code	data byte 0	data byte 1	data byte 2	data byte 3
7D0h	node	0	0Bh	x	0	0	0	0

node ID:	CAN node ID (*node*)
data type:	NODATA (*00h, 0d*)
service code:	NIS (*0Bh, 11d*)
message code:	New node ID ($1 \leq X \leq 255$)
message data:	0

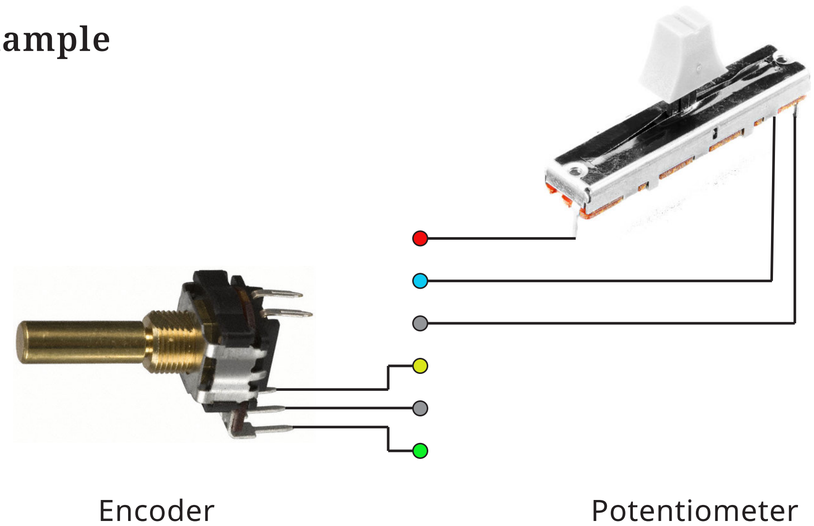
Upon completion of the Node Identifier Setting request, the response message will have a message code of **0** (zero) if the operation was successful.

Board Layout



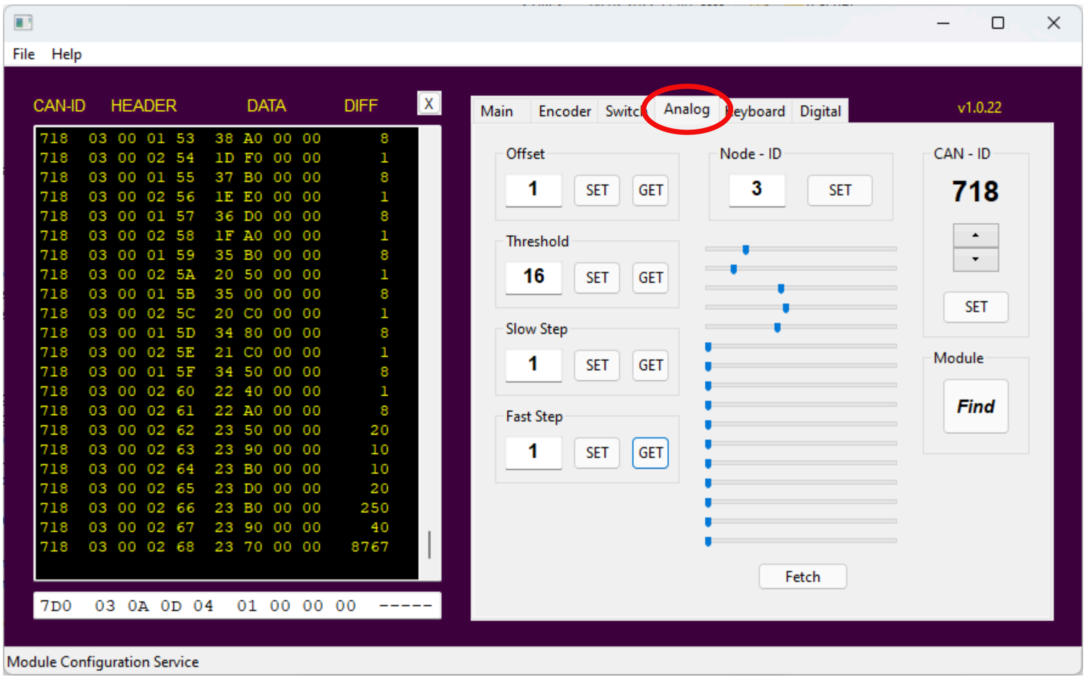
The 120 Ω jumper places a termination resistor between the CAN high and CAN low line.

Wiring Example



The Configuration Tool

The configuration tool displays an 'Analog' panel on the right side, featuring the following layout:



When you open this panel for the first time, it automatically searches for an Analog Board on the CAN bus. If a board is found, its *Node-ID*, *CAN-ID*, and *Offset* parameter are displayed. Subsequently, clicking the **Find** button initiates a new search.

Use the **GET** buttons to retrieve the *Threshold, Slow Step* and *Fast Step* parameters from the board.

All parameters, except for *CAN-ID*, can be adjusted by editing the values in their respective fields. To update these parameters in the module, press the **SET** button. For *CAN-ID*, use the spin buttons to adjust the value, then press the **SET** button to apply the changes.

To view the current status of all 16 inputs, simply press the **Fetch** button, and the horizontal sliders will update accordingly. Any activity of the board's inputs is also displayed directly by the sliders.

To simulate this activity and transmit a corresponding message on the CAN bus, simply adjust the slider pointer using the mouse.

You can monitor all CAN bus activity in the left window.

Board Dimensions [mm]