ETConcept
Engenharia de Sistemas Lda

Wiegand Converters Series

User's Manual

Models: W2RS232, W2USB and W2RS485

Document Version: 2.0

2014

Table of Contents

1. Introduction

The Wiegand converter series provides an easy solution to interconnect control access equipments. This manual describes the devices interfaces, installation and configuration.

The Wiegand converter series include the following models:

- W2RS232: Wiegand to RS232 converter;
- W2RS485: Wiegand to RS485 converter without isolation;
- W2USB: Wiegand to USB converter.

This chapter covers the following topics:

- Overview
- Features
- Package Checklist
- Specifications
- LED Indicators

1.1 Overview

The Wiegand converter series provides an easy solution to connect control access equipments, such as, keypads and card readers, to PC based controllers through RS232, RS485 or USB ports. This family of bidirectional Wiegand converters can convert data in binary and ASCII format. The converter's setup is reduced to minimum for rapid installation.

All models support Wiegand frames up to 96 bits long. The Wiegand timings, frame format and output format can be configured and stored in the device EEPROM using the configuration application .

The W2USB model is bus-powered, not requiring an external power supply. Drivers are available which allow the device to work with Windows, Linux and Mac operating systems. The driver creates a Virtual COM port (VCP) allowing the USB device to appear as a COM port available to the PC.

The W2RS485 model can be used as a Wiegand extender allowing distance of up to 1200 meters.

1.2 Features

The Wiegand converter has the following features:

- Bidirectional conversion
- Automatic detection of the direction based on lines levels
- Wiegand frames up to 96 bits
- Two general purposed I/O
- Tamper signal forwarding
- Bus powered (W2USB model only)
- Support for RS-422 and 4-wire RS-485 (W2RS485 model only)
- Configurable prefix and suffix chars to send before and after card data

1.3 Package Checklist

The following items are included in the package:

- Wiegand Converter
- Quick installation Guide
- Warranty Statement

1.4 Specifications

Table 1.1: Specifications

Wiegand Interface	
Frame Length	6 - 96 bits
Pulse Width	37 - 200 μs
Bit Period	1 - 2 ms
Idle Period	Min. 30 ms
General Purpose I/O	2
Tamper Signal	1
RS232 Serial Interface	
Ports	1
Signals	TxD, RxD, GND
Connector	Male DB9
RS485 Serial Interface	
Ports	1
Signals	TxD+, TxD-, RxD+, RxD-, GND
Connector	Terminal block
USB Interface	
Ports	1
Connector	USB type B
Serial Communication Parameters	
Speed	9600 Bauds
Data bits	8 bits
Stop bits	1 bits
Parity	None
Flow control	None
Power Requirements	
Supply voltage	7 - 22 V
Power Consumption	150 mA
Output Power (W2USB model only)	12 V and 100 mA
Mechanical	
Dimensions	55 x 50 x 25 mm (L x W x H)
Material	Anodized aluminum
Weight	200 g
Environmental	
Operating Temperature	0ºC to 55ºC (32ºF to 131ºF)
Storage Temperature	-20ºC to 70ºC (-4ºF to 185ºF)
Operating Humidity	5% to 95% of RH
Warranty	
	2 Years

1.5 Front panel layout

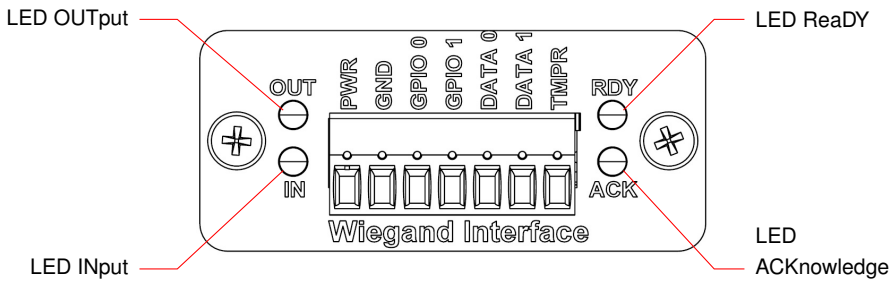


Figure 1.1: Front panel

1.5.1 LED Indicators

The LED position in the Wiegand interface panel are shown in figure 1.1 and the respective functions descriptions are in table 1.2.

Table 1.2: LED description

Name	Color	Description
INput	Green	On: Convert wiegand to serial Off: ——
OUTput	Green	On: Convert serial to wiegand Off: ——
ReaDY	Yellow	On: Ready to receive data Off: Power off or processing received data
ACK	Red	On: Data received Blink: Error occurred (see table 1.3) Off: ——

Table 1.3: ACK LED indicator Error codes

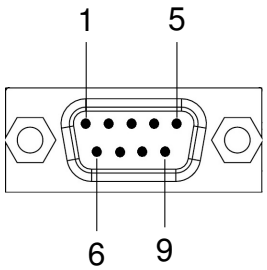
No. Blinks	Blink Rate		Error description
1	slow	-	No error
2	fast	. .	Wrong Sync. Char
2	slow	- -	Invalid command
3	fast	. . .	Receiver Timeout
4	fast	. . . .	Unknown command
5	fast	. . . .	Wrong termination char

2. Pin assignments and Wiring

2.1 W2RS232 model

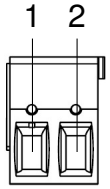
2.1.1 Serial port pin assignments

Table 2.1: RS232 Pin assignment

DB9 (male)	Pin	RS232 signal
	1	-
	2	TX
	3	RX
	4	-
	5	GND
	6	-
	7	-
	8	-
	9	-

2.1.2 Terminal block pin assignments

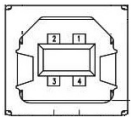
Table 2.2: Terminal Block Pin assignment

Terminal block	Pin	Signals
	1	Power Supply Positive (+)
	2	Power Supply Negative (GND)

2.2 W2USB model

2.2.1 Serial port pin assignments

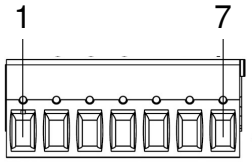
Table 2.3: USB Pin assignment

USB type B (Female)	Pin	Signals
	1	VBUS
	2	D-
	3	D+
	4	GND

2.3 W2RS485 model

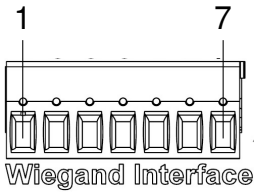
2.3.1 Serial port pin assignments

Table 2.4: RS-422/RS-485 Interface Pin assignment

Terminal block	Pin	Signals	TIA/EIA-485-A
	1	Power Positive (+)	
	2	GND	
	3	RxD-	B
	4	TxD-	Z
	5	TxD+	Y
	6	RxD+	A
	7	BGND	Common

2.4 Wiegand interface Terminal block pin assignments

Table 2.5: Wiegand Interface Pin assignment

Terminal block	Pin	Signals
	1	Power Supply Positive (+)
	2	Power Supply Negative (GND)
	3	GPIO 0
	4	GPIO 1
	5	Wiegand Data 0
	6	Wiegand Data 1
	7	Tamper signal

3. Communication

In this chapter the complete command set of the converter is described. The command format is as follows:

$Sync_1$	$Sync_2$	CMD ID	CMD Data	X	CR

where the first two chars, on the left, are the synchronization chars, followed by the command ID (CMD ID) byte and by the data bytes of the command. The X represents a reserved char, for future use, which can be set to any value, and the CR represents the terminator char, the carriage return <CR> (with value 0Dh in Hex).

The transmission order is the synchronization char, $Sync_1$, first and the carriage return <CR> last.

The synchronization chars are equal and defined with the value 55 in Hex format, i.e. $Sync_1$=$Sync_2$=55_h, which corresponds to the ASCII char 'U'.

Table 3.1: Command set summary

CMD ID		Description	
Dec	**Hex**	**Input mode**	**Output mode**
01	01_h	Received Wiegand frame	Generate Wiegand frame
02	02_h	Tamper Signal Status	Set/Clear Tamper
03	03_h	Set/Clear $GPIO_0$ Signal	$GPIO_0$ Signal status
04	04_h	Set/Clear $GPIO_1$ Signal	$GPIO_1$ Signal status
...	...	...	
09	09_h	Write to EEPROM	
10	$0A_h$	Soft Reset	
11	$0B_h$	EEPROM Dump	

3.1 Command detail

3.1.1 Wiegand frame (01_h)

The Wiegand-frame command is as follows

'U'	'U'	01_h	Nb	Wiegand raw data	t_p	t_b	X	CR

where Nb is a byte representing the number of bits of the wiegand-frame followed by the wiegand raw data within an array of twelve bytes. The t_p and t_b bytes represents, respectively, the wiegand-frame pulse width and bit period as shown in figure 3.1 on the following page. However, these bytes are not used by the converter. To set the wiegand timmings please refer to EEPROM settings. Note that the valid range of Nb is between 6 and 96, any value outside this range can yield an unpredictable result.

In output mode, see table 1.2 on page 4, the converter processes the command, copies the data to the Wiegand output buffer and sends, through the wiegand port, the specified frame.

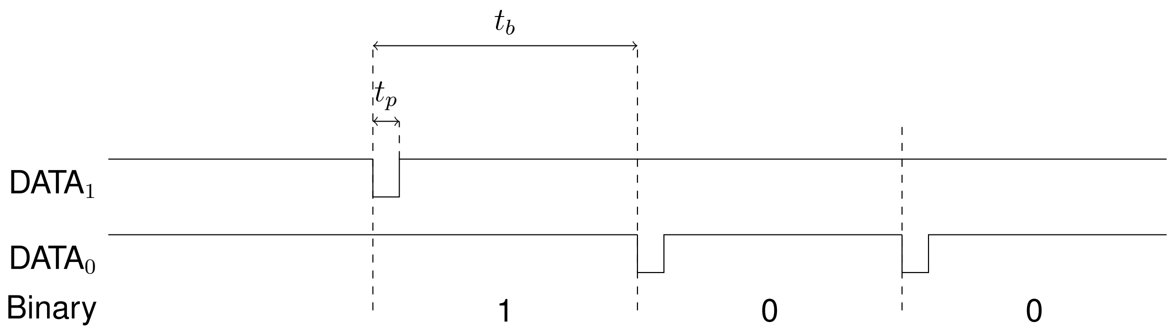


Figure 3.1: Wiegand timings

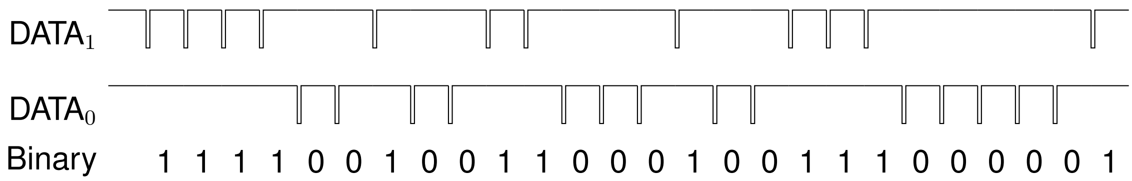


Figure 3.2: 26 bit Wiegand frame

For example, to send a standard 26 bits wiegand-frame with facility code E4$_h$ and badge id C4E0$_h$, send the command above which generates the 26 bit Wiegand-frame shown in Figure 3.2. Notice that the parity bits should be added to the raw data. For more examples see chapter **??** on page ??.

55$_h$	55$_h$	01$_h$	0A$_h$	F2$_h$ 62$_h$ 70$_h$ 40$_h$ 00$_h$ 00$_h$ 00$_h$ 00$_h$ 00$_h$ 00$_h$ 00$_h$ 00$_h$	00$_h$	00$_h$	00$_h$	0D$_h$

It is important to notice that, the converter needs a standby period, t_W, between Wiegand-frames in order to process and transmit data, as shown in figure **??** on page ??. Check the specifications of your reader, or controller, for this requirement.

3.1.2 Tamper signal (02$_h$)

The converter in input mode sets the tamper port, see table 2.5 on page 6, to input. Any state change of the port will trigger a serial transmission of a tamper signal command. In this mode, sending a tamper command to the converter has no effect on the port state and the command is discarded.

For the output working mode the converter sets the tamper port to output. The tamper command sets the port level.

The command format is as follows:

55$_h$	55$_h$	02$_h$	VALUE	00$_h$	0D$_h$

where VALUE is a byte representing the tamper port state. Value 00$_h$ represents a low-level on the tamper port and FF$_h$ a high-level.

3.1.3 GPIO$_0$ signal (03$_h$)

The converter working in output mode sets the GPIO$_0$ port, see table 2.5 on page 6, to input. Any state change of the port will trigger a serial transmission of a GPIO$_0$ signal

command. In this mode, sending a $GPIO_0$ command to the converter has no effect on the port state and the command is discarded.

For the input working mode the converter sets the $GPIO_0$ port to output. The $GPIO_0$ command sets the port level.

The command format is as follows:

55_h	55_h	03_h	VALUE	00_h	$0D_h$

where VALUE is a byte representing the $GPIO_0$ port state. Value 00_h represents a low-level on the $GPIO_0$ port and FF_h a high-level.

3.1.4 $GPIO_1$ signal (04_h)

The converter working in output mode sets the $GPIO_1$ port, see table 2.5 on page 6, to input. Any state change of the port will trigger a serial transmission of a $GPIO_1$ signal command. In this mode, sending a $GPIO_1$ command to the converter has no effect on the port state and the command is discarded.

For the input working mode the converter sets the $GPIO_1$ port to output. The $GPIO_1$ command sets the port level.

The command format is as follows:

55_h	55_h	04_h	VALUE	00_h	$0D_h$

where VALUE is a byte representing the $GPIO_1$ port state. Value 00_h represents a low-level on the $GPIO_1$ port and FF_h a high-level.

3.1.5 Write EEPROM (09_h)

This command writes a byte to the device EEPROM memory. The command has the following format:

55_h	55_h	09_h	ADDRESS	VALUE	00_h	$0D_h$

where the bytes ADDRESS and VALUE represent, respectively, the destination address and the byte to store. The destination address is limited to position 127 (see configuration EEPROM description). The higher addresses are reserved for logging the errors and warnings and, cannot be accessed by the write commando.

Note that any change to the configuration will only takes effect after reset.

3.1.6 Soft Reset ($0A_h$)

This command forces a reset. The device restarts two seconds after the receptions of this command. The command has the following format:

55_h	55_h	$0A_h$	00_h	00_h	$0D_h$

During reset the Led ACK stays on for two seconds.

3.1.7 EEPROM dump ($0B_h$)

This command dumps the EEPROM contents to the serial interface. The command has the following format:

55_h	55_h	$0B_h$	00_h	00_h	$0D_h$

3.2 EEPROM memory map

The device's EEPROM stores the configuration registers (the first 128 bytes) and the event logger (the last 128 bytes). This chapter starts with a detailed description of each configuration register. Followed by the description of the errors and events codes stored in the log.

Table 3.2 presents the device register map located in the EEPROM.

Table 3.2: EEPROM register map summary

Address		Register	Length	Default	Description
Dec	Hex	Name	Bytes	Value	
00	00	Working mode	1	$A5_h$	Working mode register
01	01	Pulse width (T_P)	2	$000D_h$	Wiegand Pulse width
02	02				
03	03	Bit Period (T_B)	2	$1CCC_h$	Wiegand bit period
04	04				
05	05	Timeout (T_W)	2	$C9FF_h$	Wiegand frame timeout
06	06				
07	07	Output Format	1	01_h	Serial output format
08	08	Log Index	1	01_h	Next log index
09	08	Baudrate	1	04_h	Baudrate index
10	0A	Data bits	1	01_h	Number of data bits idx.
11	0B	Stop bits	1	00_h	Number of stop bits idx.
12	0C	Parity	1	00_h	Parity type
13	0D				
...	...	Output format 1	16	-	Output format 1 fields
28	1C				
29	1D				
...	...	Output format 2	16	-	Output format 2 fields
44	2C				
45	2D				
...	...	Output format 3	16	-	Output format 3 fields
60	3C				
61	3D				
...	...	Output format 4	16	-	Output format 4 fields
76	4C				

Table 3.2: EEPROM register map summary (continued)

Address		Register	Length	Default	Description
Dec	Hex	Name	Bytes	Value	
77	4D				
...	...	Output format 5	16	-	Output format 5 fields
92	5C				
93	5D	Reference Level	1	10_h	Wiegand ref. level
94	5E				
...	...	Events log	128	-	Events log zone
100	64			-	

The converter supports up to 5 different ASCII output formats and, on each format, 3 different fields can be configured. The configuration of these formats are stored in the output format registers, as shown in table 3.3.

Table 3.3: Output format registers

Rel. Address		Register	Default	Description
Dec	Hex	Name	Value	
+ 00	00	Nbits	00_h	Wiegand-frame number of bits
+ 01	01	Spos	-	Wiegand $Field_1$ position
+ 02	02	Slen	-	Wiegand $Field_1$ length
+ 03	03	Fpos	-	Wiegand $Field_2$ position
+ 04	04	Flen	-	Wiegand $Field_2$ length
+ 05	05	Bpos	-	Wiegand $Field_3$ position
+ 06	06	Blen	-	Wiegand $Field_3$ length
+ 07	07	PePos	-	Parity even bit position
+ 08	08	PeBegin	-	Parity even start bit
+ 09	09	PeEnd	-	Parity even stop bit
+ 10	0A	PoPos	-	Parity odd bit position
+ 11	0B	PoBegin	-	Parity odd start bit
+ 12	4C	PoEnd	-	Parity odd stop bit
+ 13	4D	Prefix	'<'	Prefix added to each field
+ 14	0E	Suffix	'>'	Suffix added to each field
+ 15	0F	Flags	FF_h	Output format flags

3.2.1 Registers

Working mode register

The value of the register defines the device working mode. See the options in table 3.5 on page 13.

Table 3.4: Working mode register

Register Value	Description
$A5_h$	Converter in automatic detection
$0F_h$	Converter configured to input mode
$F0_h$	Converter configured to output mode

Pulse width

The value of the register defines the wiegand pulse width, t_p. Use equation (3.1) to determine t_p in µs.

$$t_p \approx 2.8836 \times (\mathsf{T_P}) \tag{3.1}$$

Bit Period

The value of the register defines the wiegand period. Use equation (3.2) to determine t_b in µs.

$$t_b \approx 0,2713 \times (\mathsf{T_B}) \tag{3.2}$$

Timeout

The value of the register, $\mathsf{T_W}$, defines the timeout period, t_w, to assume the end of wiegand frame. The timeout only affects the receiving of wiegand frames (INPUT mode). Use equation (3.3) to determine the timeout in µs.

$$t_w \approx 0,2713 \times (65535 - \mathsf{T_W}) \tag{3.3}$$

The device measures the timeout period using a 16 bit internal timer that increments on every clock cycle of the processor. The timeout timer starts after the first wiegand pulse and on every wiegand pulse its value is reset to $\mathsf{T_W}$. At the end of the wiegand frame no further pulses are received and the timer reaches the value 65535 which triggers the wiegand frame processing, see the timing diagram in Figure 3.3.

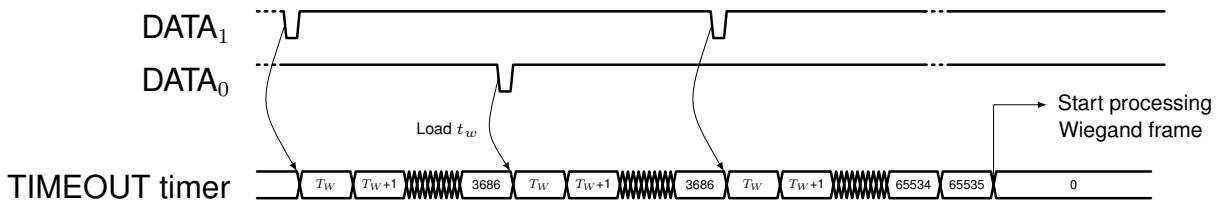


Figure 3.3: Timeout timings

Output Format

The value of the register is an index of table 3.5 on the next page which defines the output format.

Table 3.5: Working mode register

Register Value	Description
01_h	Binary output
02_h	ASCII output

Log Index

The value of the register represent the address were it will be logged the next event or error. The register value is automatically updated after a log and maintained within the range of 128 to 255.

Baudrate

The value of the register is an index of table 3.6 which defines the communication baudrate.

Table 3.6: Baudrate

Register Value	Baudrate
00_h	300
01_h	1200
02_h	2400
03_h	4800
04_h	**9600 (default)**
05_h	19200
06_h	38400
07_h	57600
08_h	115200
09_h	Reserved

Data bits

The value of the register is an index of table 3.7 which defines the communication data bits.

Table 3.7: Data bits

Register Value	Data bits
00_h	7 bits
01_h	**8 bits (default)**

Stop bits

The value of the register is an index of table 3.8 which defines the communication stop bits.

Table 3.8: Stop bits

Register Value	Stop bits
00_h	**1 stop bit (default)**
01_h	2 stop bits

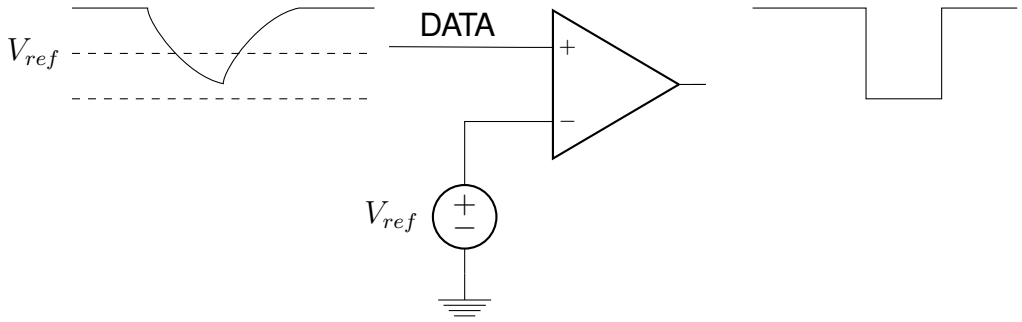


Figure 3.4: Wiegand line detector

Parity

The value of the register is an index of table 3.9 which defines one of three parity types for the communication.

Table 3.9: Parity type

Register Value	Parity type
00_h	**None (default)**
01_h	Even
02_h	Odd

Reference Level

The wiegand DATA lines levels are detected by two independent comparators with a common reference voltage, as shown in figure 3.4. By default, this value is 16 which corresponds to a 2.5V reference (half the supply voltage 5V). However, for long lines, many times, the wiegand pulse does not reach the low level, hence, the pulses are not detected. To compensate the line loss one can adjust the reference level up by changing the reference level register. The voltage reference level is given by

$$V_{ref} = 5 \times \frac{RegisterValue[4:0]}{2^5}, \tag{3.4}$$

where the less 5 significant bits of the register value are used.

Warning : Changing the reference level may lead to false detections due to noise. So when possible adjust the pulse width instead and leave the reference value unchanged!

Event log codes

The event log stores the last events, e.g. a device reset, on the high half-side of the device EEPROM. The address of the next log is defined by the *Log index* register, see section 3.2.1 on page 13. The log codes are shown in table 3.10 on the following page.

Table 3.10: Event codes

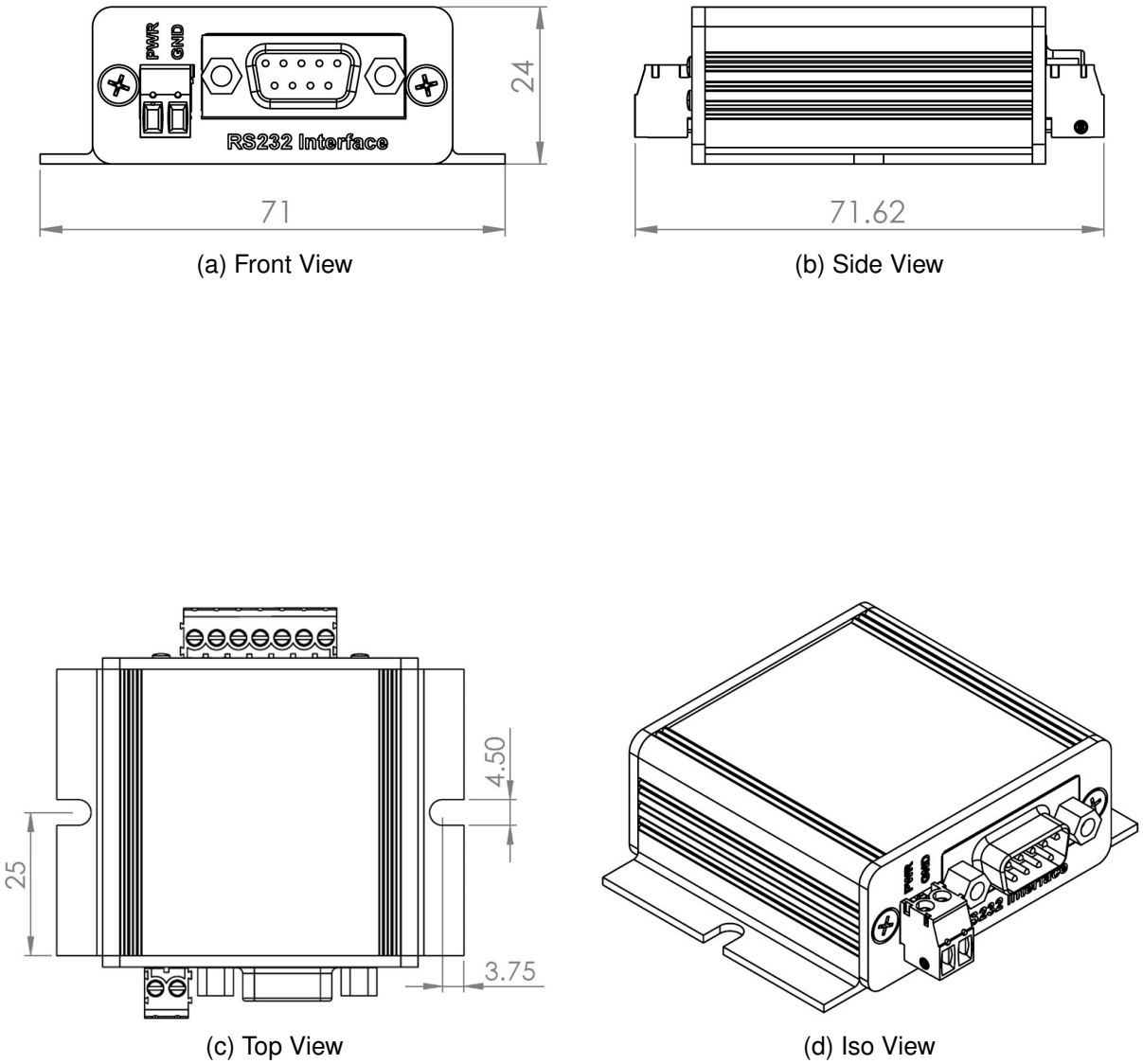
Code	Description
00_h	No error [a]
FF_h	Software reset
FE_h	Hardware reset
FD_h	Brown-out Reset (Power supply $<$ 2.4V)
FC_h	Power-up [a]
FB_h	EEPROM corruption
FA_h	Memory Error
$F9_h$	Communication Error
$F8_h$	Invalid Command
$F7_h$	Illegal Command

[a] allways followed by a reserved code

4. Mechanical Specifications

All dimensions are in millimeters.

4.1 W2RS232 model



(a) Front View

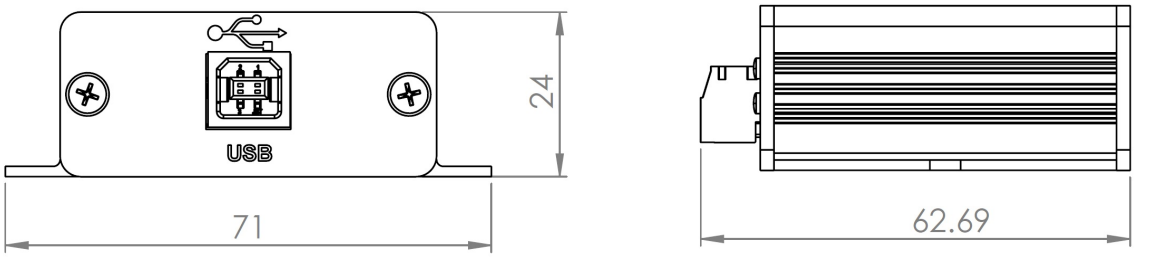
(b) Side View

(c) Top View

(d) Iso View

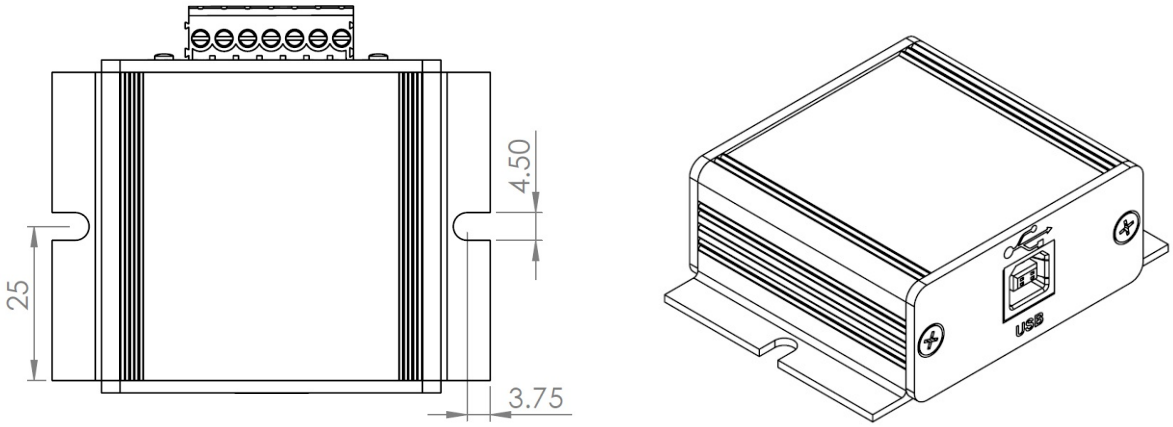
Figure 4.1: Front panel

4.2 W2USB model



(a) Front View

(b) Side View

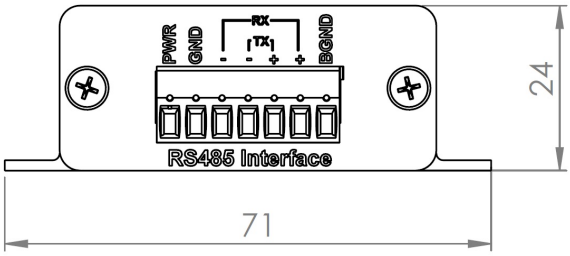


(c) Top View

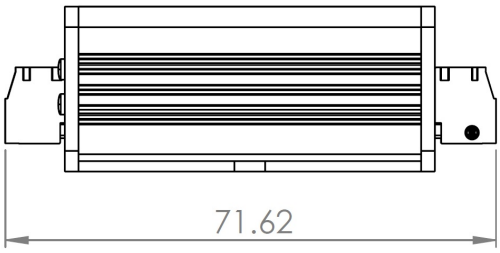
(d) Iso View

Figure 4.2: Front panel

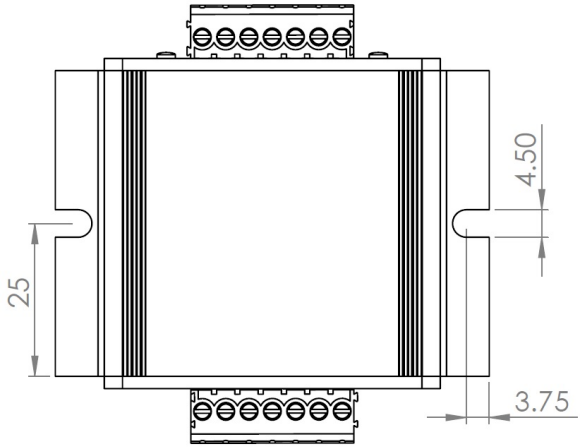
4.3 W2RS485 model



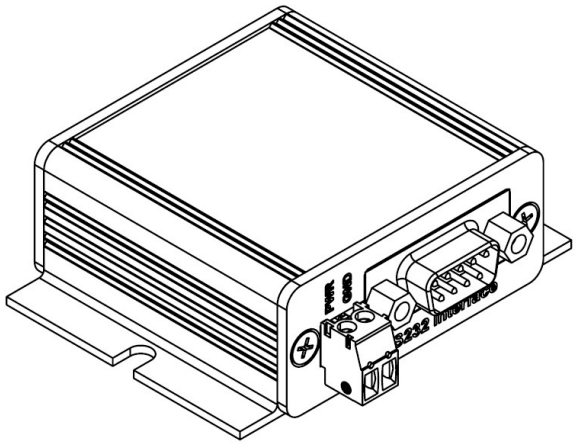
(a) Front View



(b) Side View



(c) Top View



(d) Iso View

Figure 4.3: Front panel