OIPC200-W

Wireless power control kit

General Description

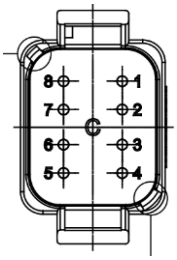
OIPC200-W is a smart electronic wireless kit (tx+rx) with 2.4GHz interface and CAN bus interface.
The kit is composed of a receiver unit (OIPC200-WRX) that, combined with a remote control (OIPC200-WTX), acquires inputs through wireless interface at 2.4 GHz.
These inputs are then available on vehicle bus, through a CAN message. The message can be used to actuate some functions in the machine.
The OIPC200-W is a versatile, smart and very cheap device, that extends the network in the machine using 2.4 GHz worldwide standard. The link has been designed to cover about 50m in environment with the presence of obstacles.
OIPC200-W can detect two different digital inputs, but it can be also extended to more switches or different types of input.
Components are protected from moisture and dust.

Applications

- Power extension in mobile machines
- SMART controllers for special tools
- Wireless control units
- Cable replacement
- Retro fit functions on special vehicles

Features

- **Low cost**
- **Simple and reliable**
- **Universal 2.4GHz frequency (worldwide)**
- **Integrated antenna**
- **Compact transmitter**
- **IP67 receiver**
- **CAN Output frame with status indication**
- **Bootloader available for firmware updates**



Power and output connector

Ordering Information

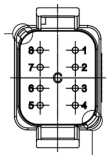
OIPC200-WRX	Receiver unit with CAN output
OIPC200-WTX	Transmitter unit, 2 buttons

TECHNICAL CHARACTERISTICS RX PART

T_A = 25°C, unless otherwise noted

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
T_A	Operating Temperature Range		-20		70	
V_{SUPPLY}	Supply Voltage Range	-20 < T_A < +70	10	12/24	30	V
I_{SUPPLY}	Current consumption			30	50	mA
N_{DIN}	Number of Digital Inputs	Wireless		2		-
N_{CAN}	Number of CAN interfaces			1		-
N_W	Number of wireless Interface	2.4GHz		1		-

RX INPUT POWER AND CAN AND CONNECTOR (GREEN)

Name	Symbol	No	AT04-08PC	No	Symbol	Name
Config Mode In	CI	8		1	CO	Config Mode Out
Not connected		7		2		Not connected
Ground	GND	6		3	+Vcc†	Positive battery power supply
CAN Low	CANL	5		4	CANH	CAN High

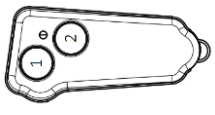
TECHNICAL CHARACTERISTICS TX PART

T_A = 25°C, unless otherwise noted

Tx must be equipped with CR2032 battery or equivalent

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
T_A	Operating Temperature Range		-20		50	
V_{SUPPLY}	Supply Voltage Range	-20 < T_A < +50	2.5	3	3.3	V
I_{SUPPLY}	Peak current consumption	Tx mode continuous			100	mA
P_M	Power consumption (max)	V_S=2,7V Tx always ON		2,75		mA*h
P_{AVG}	Power consumption (normal operation)	V_S=2,7V duty = 5% ON		135		uA*h
B_D ‡	Battery duration	CR2032 duty < 5% ON		1500		h
N_{DIN}	Number of Switches			2		-
N_W	Number of wireless Interface	2.4GHz		1		-

TX FUNCTIONS

Name	Symbol	No	BUTTONS	No	Symbol	Name
Button 1 (function1)	B1	1		2	B2	Button 2 (function 2)

† Vcc terminal is fused with 2A mini fuse, located internally, close to led indications

‡ Battery duration is estimated considering the use of a CR2032 battery - 220mAh capacitance – ambient temperature - remote usage equivalent of 1 hour per day

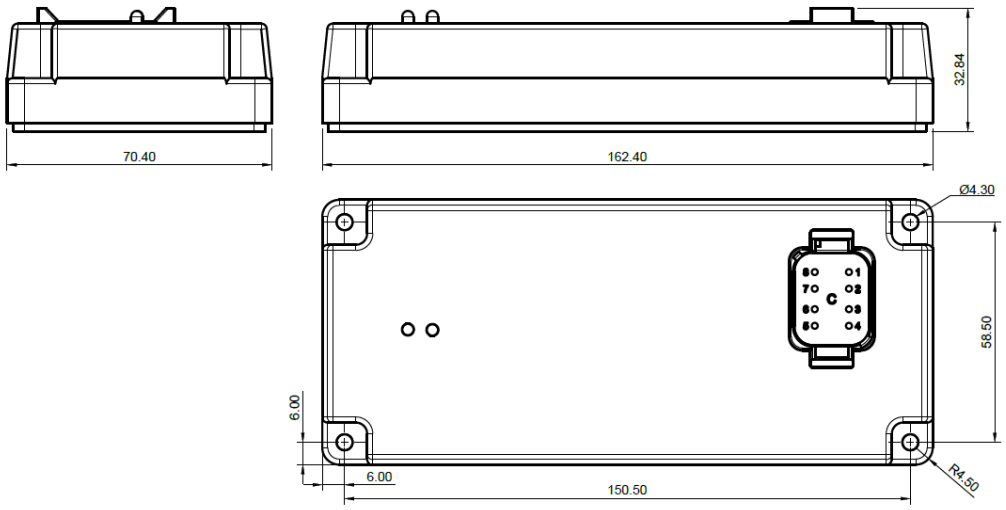
MECHANICAL CHARACTERISTICS AND DIMENSIONS

Tolerance: ±0.25mm unless otherwise noted

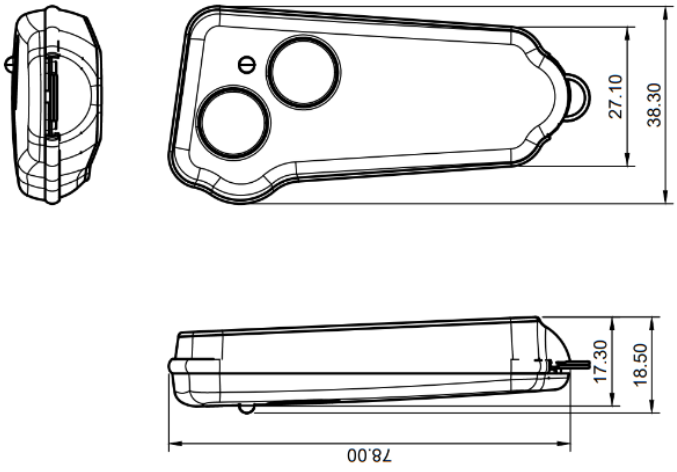
Symbol	Parameter	Conditions	Min	Typ	Max	Unit
W_R	Width rx			70,4		mm
L_R	Length rx			162,4		mm
H_R	Height rx			37,7		mm
W_R	Weight rx			250		g

W_T	Width tx			38,3		mm
L_T	Length tx			78		mm
H_T	Height tx			17,3		mm
W_T	Weight tx			25		g

DRAWINGS RX



DRAWINGS TX



OIPC200-WRX SHORT MANUAL

A general knowledge of CAN OPEN protocol is recommended to fully understand this manual. All the numbers contained are written in decimal format. Hexadecimal values are marked with a final "h" to distinguish them from decimal.

This document can be subjected to changes without notice and no claims can be derived from its details, illustrations, or descriptions.

CANBUS connections and termination resistors

The OIPC200-WRX receiver doesn't have internal bus line terminator resistor. The user must ensure two 120Ω termination resistors are installed between the CANH and CANL lines. Typically, one is located near the network master at the start of the bus. The other should be at the end of the CANBUS, on the furthest node of the CAN network.

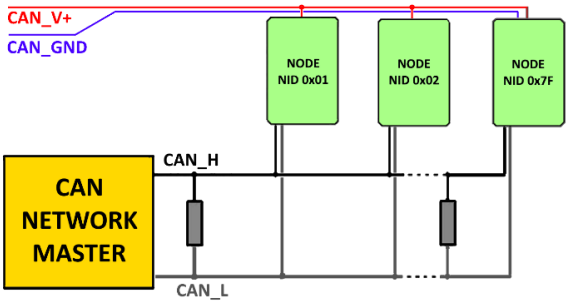


Figure 1 – CAN BUS network typical wiring scheme

CANOPEN

CANopen (EN 50325-4) is a high-level CAN-based communication protocol, developed for embedded networking applications (e.g.: in-vehicle networks). CANopen comprises and standardizes many aspects of a communication protocol like network programming framework, device descriptions, interface definitions and application profiles, enabling different devices and applications from different manufacturers to communicate.

OIPC200-WRX receiver meets CiA 301 v.4.2.0 requirements for a general-purpose CANopen network node. No device profile is specified for wireless receiver.

Supported communication objects

OIPC200-WRX only supports CAN frames with 11-bit node identifiers. The first 7-bit (less significant) of the 11-bit communication object identifier (COB-ID) specifies the node identifier value (NID) while the remaining 4-bit (the most significant) specifies the function code. During the initialization phase, all the different COB-ID identifiers values are computed starting from the dictionary object values.

Communication object	Direction [§]	COB – ID	Object description
NMT	RX	00h	NMT services
SYNC	RX	80h	Sync object
EMCY	TX	80h + NID	Emergency object
TPDO1	TX	180h + NID	Remote buttons and status register
SDO (Client→Server)	RX	600h + NID	Access to a node object dictionary
SDO (Server→Client)	TX	580h + NID	Node reply to SDO request
Boot Up \ Heartbeat	TX	700h + NID	Boot Up and Heartbeat messages

Table 1 –Supported objects

[§] Direction is considered from the point of view of OIPC200-WRX. CANopen SDO standard communication protocol defines the device on which the object dictionary table resides as master device.

Boot-Up frame

The OIPC200-WRX initialization process starts as soon as the device is powered on. This process takes around 1 second to finish. As soon as initialization is complete, the device sends a boot-up frame (see Table 2) to advise it has entered the pre-operational state.

COB-ID	Byte 0
700h + NID	00h

Table 2 –Boot-Up message frame

NMT finite state machine

A CANopen device's behavior depends on the state of a *finite state machine* (see Figure 2). Each state defines the node's behavior, the valid communication objects and the possible actions.

After the Boot-Up frame is transmitted, the **OIPC200-WRX receiver automatically enters the NMT operational state and begins to send a TPDO1 data frame every 100ms.**

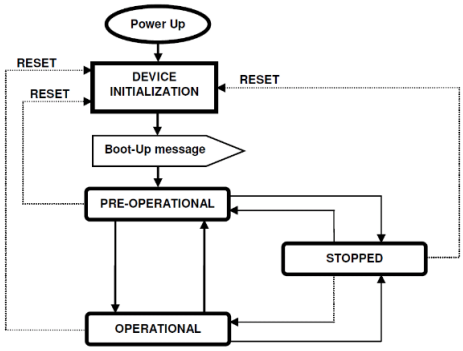


Figure 2 – NMT state machine of CANopen nodes

NMT command frames (refer to *Table 3*) are used to change the active state of the NMT finite state machine. The NMT command frames are made by two data bytes. The first byte of the NMT command code (refer to *Table 4*) specifies the command itself, while the second byte indicates whether the command applies to all nodes in the network or just a specific one. If the value of the second data byte (byte 1) is 00h, the NMT command frame is sent to all nodes in the network. Otherwise, it specifies the address of the individual node to which the command applies.

A node in the *pre-operational* state, can transmit the heartbeat messages (when this service is configured), but PDO communication is not available. If a device is in the pre-operational NMT state, the SDO protocol must be used to configure or read data. Only nodes in the *operational* state support all available communication objects: PDO communication objects are only available in the *operational* state. Devices in the *stopped* state only process NMT commands. A *stopped* device indicates its current NMT state by supporting the error control protocol (heartbeat).

COB ID (11bit)	Byte0	Byte1	
000h	COMMAND CODE (see Table 4)	00h	broadcast NMT command to all the nodes of the network
		NID	send NMT command to a specific node

Table 3 – NMT command frame organization

COMMAND CODE	EFFECT ON NMT STATE MACHINE
01h	Go to OPERATIONAL
02h	Go to STOPPED
80h	Go to PRE – OPERATIONAL
81h	Go to RESET NODE
82h	Go to RESET COMMUNICATIONS

Table 4 – NMT command code values

PDO - Process Data Object

PDO messages can only be transmitted when the NMT state machine is in the *operational* state (refer to section 0). This type of messages consists solely of process data (e.g.: remote button status, …). PDO message frames are ideal for the cyclical exchange of process data. With the PDO frames all the additional fields required by the SDO messages: index, subindex and data length. This allows the PDO message to contain only the useful data with no overhead. The OIPC200-WRX receiver supports a single *Transmit Process Data Object* (TPDO1).

TPDO1 frame

By default, the OIPC200-WRX receiver has no input and no output connections: it receives the configuration from its transmitter and sends a frame over can network every 100ms. The remote buttons and status register is included in the data field of the TPDO1 frame. The remote buttons and status register is cleared 100ms after the reception of the last radio message transmitted by the remote.

COB ID	Byte0
0x180 + NID	Remote button and status register

Table 5 – TPDO1 frame structure

TPDO1 DATA BYTE 0							
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
		OIPC200-WRX configuration mode active	OIPC200-WRX has one paired OIPC200-WTX			OIPC200-WTX Button 2	OIPC200-WTX Button 1

Table 6 – Structure of the TPDO1 byte 0 and the meaning of individual bits (empty bytes are not used, and their value is always zero)

TPDO1 FRAMES EXAMPLES:

COB-ID	Byte 0	REMOTE PAIRED	REMOTE BUTTON STATE
0x1E4 (NID = 64h)	00h	NO	No data received from remote
	01h		Button #1 pressed
	02h		Button #2 pressed
	03h		Button #1 + Button #2 pressed → PAIRING REQUEST
	10h	YES	No data received from remote
	11h		Button #1 pressed
	12h		Button #2 pressed
	13h		Button #1 + Button #2 pressed [**]

Table 7 – Example of TPDO1 ftames

TPDO TRANSMISSION TYPES

The scheduling of the TPDOs frames is based on the transmission type sub-index (02h) of the TPDO communication parameters objects at index 1800h. The table below lists all the possible TPDO transmission types for a generic CANopen device and indicates which of these are available for the OIP200-WRX receiver.

Values	Description	Supported
00h	Acyclic synchronous: triggered if SYNC is received and one of the mapped process values has changed after the last transmission	NO
01h…F0h	Cyclic synchronous: triggered when the number of SYNC received matches the values of the transmission type object [††]	YES
F1h…FBh	Reserved	-
FCh	Synchronous RTR only: not recommended	NO
FDh	Asynchronous RTR only: not recommended	NO

(**)WARNING: If both buttons are held pressed for 10 seconds, the remote will reset its configuration to the default settings. The pairing with the receiver will be canceled.
(††) e.g.: with transmission type sub-index equal to 01h the TPDO is cyclic every 1 received SYNC; with 02h the TPDO is cyclic every 2 received SYNC; …; with F0h the TPDO is cyclic every 240 received SYNC

FEh	Asynchronous: triggered by one or more manufacturers defined internal events. e.g.: the TPDO1 event timer expire	YES
FFh	Asynchronous: triggered by one or more internal events defined in the device profile and the application profile	NO

Table 8 – TPDO transmission types

SDO Read

The read operation for a device's object dictionary entry starts with an SDO upload request frame. The frame first data byte (byte 0) indicates the data type of the entry to be read from the dictionary.

COB ID	Byte0	Byte1	Byte2	Byte3
600h + NID	RD data type [‡‡] (Table 10)	Index Low (LSB)	Index High (MSB)	Sub-Index

Table 9 – organization of an SDO read request frame.

OD entry type	RD data type
Read any length	40h
1 byte	4Fh
2 bytes	4Bh
4 bytes	43h
STRING	41h

Table 10 – possible RD data type values for SDO read request frame.

A successful reading operation returns the object's value in the response frame. The value can be an 8-bit, 16-bit, or 32-bit number or 1 to 4 ASCII characters for string-type entries. If the object's value is shorter than 4 bytes, the unused bytes are omitted, resulting in a shorter response frame. The 16-bit and 32-bit object's values are returned starting from the least significant byte (LSB) to the most significant byte (MSB).

COB ID	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
580h + NID	Data type (Table 10)	Index Low (LSB)	Index High (MSB)	Sub-Index	Entry Value (LSB to MSB)			

Table 11 – organization of a SDO read response data frame.

SDO Write

The write operation to a device object dictionary entry starts with an SDO download request frame. The first data byte (byte 0) indicates the data type to be written.

COB ID	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
600h + NID	WR data type [§§] (Table 13)	Index Low (LSB)	Index High (MSB)	Sub-Index	Write Value (LSB to MSB)			

Table 12 – organization of a SDO write request frame.

The value can be an 8-bit, 16-bit, or 32-bit number, fitting into the value field (bytes 4 to 7), and must match the object dictionary entry type. If the value is shorter than 4 bytes, transmitting unused bytes is unnecessary. To write 16-bit and 32-

‡‡ If the entry data type isn't known, it's possible to use the "Read any length" (40h) data type in the SDO upload frames.
§§ If the entry data type isn't known, it's possible to use the "Write any length" (22h) data type in the SDO download frames.

bit value to an object, the SDO frame's data bytes must be written starting from the least significant byte (LSB) to the most significant byte (MSB).

OD entry type	WR data type
Write any length	22h
1 byte	2Fh
2 bytes	2Bh
4 bytes	23h

Table 13 – possible WR data type values for SDO write request frame

An error-free write operation is confirmed by an SDO response frame: byte 0 equals 60h, indicates the successful writing of the new value. The index and sub-index values are repeated in the response, and bytes 4 to 7 are all set to zero.

COB ID	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
580h + NID	60h	Index Low (LSB)	Index High (MSB)	Sub-Index	00h	00h	00h	00h

Table 14 – organization of a SDO download response data frame.

SDO ERRORS CODES

When an SDO read or write operation fails the OIPC200-WRX answers with an SDO abort frame. The first data byte (byte 0) of the abort frame is set to 80h to signal an error condition. The index and the sub-index values are included in the abort frame to specify which entry has encountered the error. The value field of the abort frame contains the 4-byte long SDO abort code, that identifies the reason for the error (refer to Table 16).

COB ID	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
580h + NID	80h	Index Low (LSB)	Index High (MSB)	Sub-Index	SDO abort code (Table 16) (LSB to MSB)			

Table 15 – organization of a SDO download request frame.

SDO abort code	Bytes 4 to 7	Error
0504 0001h	01h; 00h; 04h; 05h	SDO command not valid or unknown
0601 0002h	02h; 00h; 01h; 06h	Attempt to write a read-only object
0602 0000h	00h; 00h; 02h; 06h	Object does not exist in the object dictionary
0604 0043h	43h; 00h; 04h; 06h	General incompatibility reason
0607 0010h	10h; 00h; 07h; 06h	Data type does not match; length of service does not match
0609 0011h	11h; 00h; 09h; 06h	Sub-index does not exist
0609 0030h	30h; 00h; 09h; 06h	Parameter invalid value (download only)
0609 0031h	31h; 00h; 09h; 06h	Value of written parameter too high
0609 0032h	32h; 00h; 09h; 06h	Value of written parameter too low
0800 0000h	00h; 00h; 00h; 08h	General Error
0800 0020h	20h; 00h; 00h; 08h	Data cannot be transferred or stored to/in the application
0800 0024h	24h; 00h; 00h; 08h	No data available

Table 16 – SDO abort codes list for SDO abort frames

OBJECTS DICTIONARY

Index	Sub-index	Parameter description	Data Type	Access	Default	Range	NVM Saved
1000h	00h	Device type (device profile unknown)	UNS32	RO	00000000h		
1001h	00h	Error register	UNS8	RO	00h		
1002h	00h	Manufacturer error register	UNS32	RO	00h		
1003h	Pre-defined error field						
	00h	Largest supported sub-index	UNS8	RO	05h		
	01h	error[0]	UNS32	RO	00h		
	02h	error[1]	UNS32	RO	00h		
	03h	error[2]	UNS32	RO	00h		
	04h	error[3]	UNS32	RO	00h		
	05h	error[4]	UNS32	RO	00h		
1005h	00h	COB-ID Sync object	UNS32	RW	80h	01h...7FFh	YES
100Ah	00h	Manufacturer software version	UNS32	RO	{rev.dep.}		
100Ch	00h	Guard time [ms]	UNS16	RW	00h	0001h…FFFFh	YES
100Dh	00h	Life time factor	UNS8	RW	00h	01h…FFh	YES
1010h	Save parameters						
	00h	Largest supported sub-index	UNS8	RO	01h		
	01h	Save ALL parameters ("save" = 73617665h)	UNS32	RW	01h		
1011h	Restore all parameters						
	00h	Largest supported sub-index	UNS8	RO	01h		
	01h	Reload ALL DEFAULT parameters ("load" = 6C6F6164h)	UNS32	RW	01h		
1014h	00h	COB-ID Emergency object	UNS32	RO	80h + NID		
1017h	00h	Producer Heartbeat interval time [multiple of 1ms, 0 = disabled]	UNS16	RW	00h	0000h...FFFFh	YES
1018h	Identity object						
	00h	Largest supported sub-index	UNS8	RO	04h		
	01h	Vendor ID	UNS32	RO	00h		
	02h	Product code	UNS32	RO	00h		
	03h	HW Revision	UNS32	RO	00h		
	04h	Serial number and lot number	UNS32	RO	00h		
1200h	Server SDO1 parameters						
	00h	Largest supported sub-index	UNS8	RO	02h		
	01h	COB-ID Client > Server	UNS32	RO	600h + NID		
	02h	COB-ID Server > Client	UNS32	RO	580h + NID		

1800h	TPDO1 communication parameters						
	00h	Largest supported sub-index	UNS8	RO	05h		
	01h	COB ID	UNS32	RO	180h + NID		
	02h	Transmission type [synchronous, manufacturer specific]	UNS8	RW	FEh	00h...FEh	YES
	04h	Reserved	UNS16	RW	00h		
	05h	Event timer for cyclical transmission [multiple of 1ms, 0 = disabled]	UNS16	RW	00h	0000h...FFFFh	YES
1A00h	TPDO1 mapping parameters (fixed mapping)						
	00h	Largest supported sub-index	UNS8	RO	04h		
	01h	Remote buttons status	UNS8	RO	0x40020008		
2000h	00h	Node ID	UNS8	RW	64h	01h...7Fh	YES
2001h	00h	Baud Rate [kBit\s]	UNS16	RO	01h	[0] – 125kbps [1] – 250kbps [2] – 500kbps [3] – 1000kbps	YES
2002h	00h	Restore defualt Node ID	UNS32	RW			
4002h	00h	Remote buttons and status register	UNS8	RO	00h	see TPDO1 paragraph	
4003h	00h	Remote battery	UNS8	RO		00h…FFh	
4004h	00h	Receiver radio MAC-ID	UNS32	RO			
4005h	00h	Paired remote radio MAC-ID	UNS32	RW	0xFFFFFFFF		YES
4006h	00h	Enable broadcast reception	UNS8	RW	00h	[00h] OFF [01h] ON	YES
5002h	00h	Boot Delay Timer [multiple of 1ms]	UNS16	RW	32h	32h…7530h	YES
5003h	00h	CANBUS error control register	UNS8	RW	00h	00h…07h	YES
5004h	00h	Auto operational mode	UNS8	RW	01h	[00h] OFF [01h] ON	YES

Table 17 – Object dictionary

Save All – 1010h

This object initiates the process of storing device configuration parameters in its permanent memory. The function for saving all parameters is available. The permanent memory writing process begins only when a specific signature is written to sub-index 01h. This signature is "save," represented in hexadecimal as 65h 76h 61h 73. When the non-volatile memory writing process ends, the device signals the success of the saving process with an SDO response frame (or an SDO abort frame in case of error).

Reload All – 1011h

This object initiates the process for restoring the device's default factory configuration. The function to restore all the parameters is available. The permanent memory writing process is initiated only when a specific signature is written to sub-index 01h. This signature is the word load, represented as 64h 61h 6Fh 65h. Upon receiving the load signature, the device restores all parameters in its permanent memory to their default factory values. When the non-volatile memory writing process ends, the device signals the success of the restoring process with an SDO response frame (or an SDO abort

frame in case of error). It is strongly recommended to reset the sensor by performing a power cycle after the default values have been restored.
The reload all command does not affect the node identifier and the bit rate settings. Any changes to the node identifier and bit rate must be made through an explicit SDO write request frame. To make the new values effective, they must first be stored in permanent memory, after which the device will reboot.

Remote buttons and status – 4002h

The OIPC200-WRX receiver does not have any physical input or output connection lines. The input lines consist of the radio messages transmitted by the OIPC200-WTX remote. The remote buttons and status register provide the data about the button and pairing statuses of the transmitting remote. This register is cleared 100ms after the reception of the last radio message transmitted by the remote. The remote and status register value is available also in the TPDO1 message frame, transmitted every 100ms (default value).

Remote buttons and status register							
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
		OIPC200-WRX configuration mode active	OIPC200-WRX has one paired OIPC200-WTX			OIPC200-WTX Button 2	OIPC200-WTX Button 1

Table 18 – Meaning of the individual bit composing the remote buttons and status register (empty bytes are not used, and their value is always zero)

Remote Battery – 4003h

The remote battery register keeps track of the battery level for the OIPC200-WTX that last transmitted a radio message. The value of FFh indicates that the remote's CR2032 battery is fully charged at its nominal voltage of 3 Volts. When the battery level falls below 2.5 Volts, the remote will signal a low battery condition. The register value that corresponds to 2.5 Volts is D5h.
The battery level register shows the charge of the OIPC200-WTX battery that last transmitted a radio message, whether the remote is paired or not. To view the battery status of a remote that is not paired, you need to enable the reception of broadcast messages. Additionally, the battery level of a remote that is not paired can also be received when the OIPC200-WRX receiver is in configuration mode and a pairing request frame is received.

Receiver MAC-ID – 4004h

This register contains the radio MAC-ID address of the OIPC200-WRX receiver. The radio MAC-ID address is a 32-bit value that identifies the receiver. This value is used by the paired remote to transmit the radio message only to the paired receiver.

Remote MAC-ID – 4005h

This register contains the MAC-ID address of the paired OIPC200-WTX remote. This value is used by the receiver to receive only the messages transmitted by the paired remote. It is also used to transmit the acknowledgement radio message only to the paired remote. If the register's value is equal to FFFFFFFFh this mean no remote is paired. The presence of a paired remote is also shown in the remote buttons and status register (and consequently in the TPDO1 frame).
The remote MAC ID can be overwritten using an SDO write request. This feature allows you to reset the value to its default unpaired state after the paired remote has been reset.

Enable broadcast reception – 4006h

This register enables the reception and processing of broadcast radio messages sent by one or more unpaired remotes. When the reception of broadcast messages is activated, the data contained within those messages is processed by the receiver. The status of the buttons on the unpaired remote is recorded in the remote buttons and status register. Bit 4 of this register is set to zero to indicate that the value originates from an unpaired remote. Additionally, the remote battery register is updated to reflect the battery level of the unpaired remote.

WARNING

This feature should only be used for testing purposes. Keeping the reception of broadcast messages enabled can be dangerous because the receiver may receive and serve radio messages from multiple unpaired remotes.

LED indicators

The device features two distinct LEDs that meet the specifications outlined in CiA DR-303-3. Green led is used as status led indicator and red led is used as error led. The tables below describe all led configurations.

STATUS LED	LED state	LED state description
¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤	OFF	The device is switched off
¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤	Single flash	The device is in Stop mode
¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤	Blinking	The device is in Pre-operational mode
¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤	ON	The device is in Operational mode

ERROR LED	LED state	LED state description
¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤	OFF	The device is in working conditions
¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤	Single flash	CAN warning limit reached
¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤	Double flash	Loss of Guarding-master detected
¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤	ON	The device is in state Bus-Off

Table 19 – Rx led indications

How to enable the OIPC200-WRX configuration mode

To start the OIPC200-WRX receiver in configuration mode, please follow these steps:

1. Turn off the power supply to the OIPC200-WRX receiver.
2. Connect pin #1 (Config Mode Out) to pin #8 (Config Mode In) on the green connector.
3. Turn on the power supply to the OIPC200-WRX receiver.

Once powered on, the OIPC200-WRX receiver will enter configuration mode. Please note that the receiver will automatically return to normal mode 15 seconds after starting in configuration mode.

While in configuration mode, the receiver will accept pairing requests from unpaired remotes. The activation of configuration mode is indicated in the remote button and the status register (transmitted with the TPDO1 frames) by setting Bit 5 of this register. Once the remote is successfully paired, the receiver returns to the normal working mode. The pairing of the remote will be notified by setting the Bit 4 of the remote buttons and status register (transmitted with the TPDO1 frames). To prevent the receiver from entering configuration mode at every startup, disconnect pin #1 from pin #8 of the green connector and leave it unconnected (floating).

WARNING

If an OIPC200-WRX receiver, which is already paired with a remote, enters configuration mode and receives a new pairing request from another OIPC200-WTX remote, it will process the latest request. As a result, the receiver will only respond to the message from the new remote, leading to a loss of communication with the previously paired remote.

OIPC200-WTX MANUAL

The OIPC200-WTX remote features two buttons and a bicolor LED indicator that displays green and red. When one of the buttons is pressed, the receiver processes the data it receives from the remote and communicates the button status on the CAN network using the TPDO1 frame. If both buttons are pressed simultaneously, the configuration functions are activated. To use these configuration functions, the OIPC200-WRX receiver must be set to configuration mode. If the receiver is operating in normal mode, the status of the buttons will be displayed in the remote's button status register, but no additional functions will be initiated.

Remote Battery

The remote operates using a single CR2032 battery. When new, the battery has a nominal voltage of 3 volts. However, when the battery voltage drops below 2.5 volts, the remote's full functionality and the transmission of all radio messages may be compromised. To indicate this low battery condition, the red LED on the remote will light up continuously when one or both buttons are pressed.

Bicolor LED indicator meaning

	GREEN	signals the transmission of the radio messages
	RED	signals the active remote functions and status by differ blink schemes

CONDITION	BICOLOR INDICATOR BLINK SCHEME
UNPAIRED TX BROADCAST	GREEN LED + RED LED Transmission repeating period 500ms
PAIRED TX UNICAST	GREEN LED Transmission repeating period 100ms
PAIRING REQUEST TX BROADCAST	RED LED blink for 3 times (5ms ON; 200ms OFF) then after 250ms broadcast transmission of the pairing request radio message (GREN LED + RED LED) Transmission repeating period 2000ms
PAIRING CONFIRM RECEIVED	RED LED blinks for 10 times (5ms ON; 200ms OFF) then sends in a UNICAST transmission the acknowledge frame to the receiver (GREEN LED)
REMOTE RESET	RED LED turn ON for 1 second
LOW BATTERY	RED LED fixed ON during remote turn ON time

Table 20 – Tx led short explanation

BUTTON-RELATED BLINKING PATTERNS OF THE BICOLOR LED INDICATOR

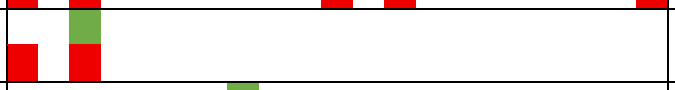
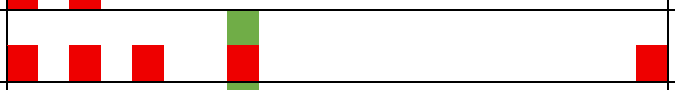
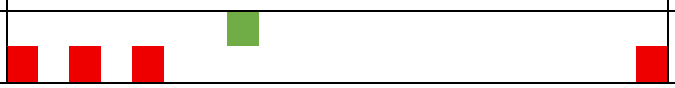
PAIRING STATUS	PRESSED BUTTON	GREEN LED [TX] RED LED [STATUS]	PERIODO [ms]	FUNCTION
UNPAIRED	S1		500	TX broadcast S1
	S2		500	TX broadcast S2
	S1+S2		1000	TX broadcast binding request
	S1+S2		1000	after 6 repetition, remote reset
PAIRED	S1		100	TX unicast S1
	S2		100	TX unicast S2
	S1+S2		1000	TX unicast binding request

Table 21 – Tx led blinking types

SPECIAL FUNCTIONS BLINKING PATTERNS OF THE BICOLOR LED INDICATOR

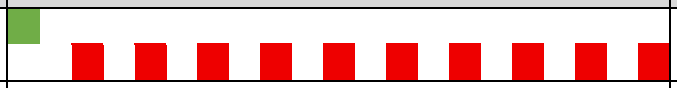
PAIRING STATUS	PRESSED BUTTON	GREEN LED [TX] RED LED [STATUS]	FUNCTION
END OF PAIRING	S1+S2		10 blinks of RED
REMOTE RESET	S1+S2		RED turns ON for 1 second
LOW BATTERY	S1 or S2		RED fixed ON until button pressed

Table 22 – Tx led blinking special functions

OIPC200-WTX PAIRING PROCEDURE

To transmit a pairing request radio message, press and hold both remote buttons simultaneously. The success of the pairing procedure depends on the active operating mode of the receiver:

1. **OIPC200-WRX operating in normal mode**
   When the OIPC200-WRX operates in normal mode, it indicates the status of the buttons and the remote battery level in the designated registers. However, no other functions will be activated. This mode occurs when the user accidentally presses both remote buttons simultaneously.

2. **OIPC200-WRX operating in configuration mode**
   When the OIPC200-WRX starts in configuration mode, there is a 15-second window to pair a remote. To pair a remote with the receiver, press and hold both remote buttons until the red indicator blinks 10 times, then immediately release the buttons. If the buttons are held down for 10 seconds, the remote will initiate the reset procedure.

WARNING

If both remote buttons are held down for 10 seconds, the remote configuration will reset to factory default values, resulting in the loss of pairing with the receiver.

OIPC200-WTX RESET PROCEDURE

To reset the OIPC200-WTX remote to its default factory settings, press and hold both buttons on the remote simultaneously for 10 seconds. This duration corresponds to approximately six repetitions of the "TX unicast binding request" blink pattern. Once the reset is complete, the remote will be unpaired and will revert to broadcast transmission mode.

WARNING

When the remote is reset to its default factory configuration, it loses its pairing with the receiver and begins to transmit in broadcast mode, using a longer repetition period of 500ms. Although the remote configuration is reset, the receiver configuration remains unchanged. The MAC-ID stored in the receiver's non-volatile memory does not revert to an unpaired value unless it is overwritten by another pairing request or through an SDO write operation on the remote MAC-ID register, or if it is restored to the default factory value. As a result, the receiver will continue to respond to frames from the unpaired remote, even though the remote has been reset and has lost its pairing information.