



OEM-DES-RFID-Lock
13.56 MHz OEM RFID Lock with CAN interface
User's Guide

iDTRONIC GmbH
Ludwig-Reichling-Straße 4
67059 Ludwigshafen
Germany/Deutschland

Phone: +49 621 6690094-0
Fax: +49 621 6690094-9
E-Mail: info@idtronic.de
Web: idtronic.de

Issue 1.2
– 11. April 2023 –

Subject to alteration without prior notice.
© Copyright iDTRONIC GmbH 2023
Printed in Germany

Contents

1	Project Overview	5
1.1	CAN-Bus Commands and Message Flow	5
1.1.1	Standard Operation of RFID-Lock	5
1.1.2	Initialising the RFID-Lock.....	5
1.1.3	Function Description Key-Operation (Standard Operation Mode).....	7
1.1.4	Function Description Key Teach-In Operation	8
1.1.5	Service Commands.....	8
1.2	Reference Documents	8
1.3	Glossary	9
2	CAN-Bus RFID Function.....	10
2.1	CAN Settings	10
2.2	KeyReader Message.....	10
2.2.1	ECU_A_Wakeup	10
2.3	UDS Protocol.....	10
2.3.1	Identifier Structure	10
2.3.2	Possible Identifier Values.....	10
2.3.3	Addresses.....	10
2.3.4	SIDs and PIDs	11
2.3.5	Response Codes	11
2.3.6	Data Identifier.....	11
3	Firmware Update via CAN-Bus	13
3.1	CAN	13
3.1.1	Identifier structure.....	13
3.1.2	Possible Identifier Values.....	13
3.1.3	Source and Destination Addresses	13
3.2	Timeout.....	13
3.3	Protocol.....	13
3.3.1	General	13
3.3.2	Data Frame	14
3.3.3	CRC Calculation	14
3.4	Commands.....	14
3.4.1	Readout the Software Version.....	14
3.4.2	Erase Flashpage	14
3.4.3	Write flash	15
3.4.4	Start application.....	15
3.5	Firmware Update Procedure	15
3.6	Application Start cycle	15
4	Electric Requirements	16
4.1	Power Supply with Overvoltage Protection.....	16
4.2	Current Consumption	16
4.3	CAN-Bus with Overvoltage Protection.....	16
4.4	Connector	16
4.4.1	Connector Information	16
4.4.2	Connector Pinout.....	16
4.4.3	Connector Components.....	17
4.4.4	Dimensions Connector Housing.....	17

4.4.5	Dimensions Crimp Contact	17
4.5	SWD connector for Firmware update of the CAN-Bus MCU	17
4.6	Hardware Information	17
5	Mechanics	18
5.1	Antenna Read Direction	18
5.2	Dimensional Drawing	18
6	Technical Data	19
7	Revision History	20

1 Project Overview

This RFID devices can work as a lock for a vehicle.

During production you can program DESFire tags to become RFID keys.

In standard mode, the RFID-Lock automatically detects a valid RFID key, wakes up the ECU and sends the key number to the ECU.

The key number is password-protected and encrypted in the DESFire filesystem.

1.1 CAN-Bus Commands and Message Flow

Communication with the vehicle is done using a 3DES encryption.

There are different Messages flows using specific CAN messages, which are described here.

1.1.1 Standard Operation of RFID-Lock

After initialisation and programming several RFID-Keys, the RFID-Lock switches to low-power mode after 5 min.

In low-power mode the RFID-Lock detects a valid RFID-Key within 1 second. After detecting a valid RFID-Key, it does this:

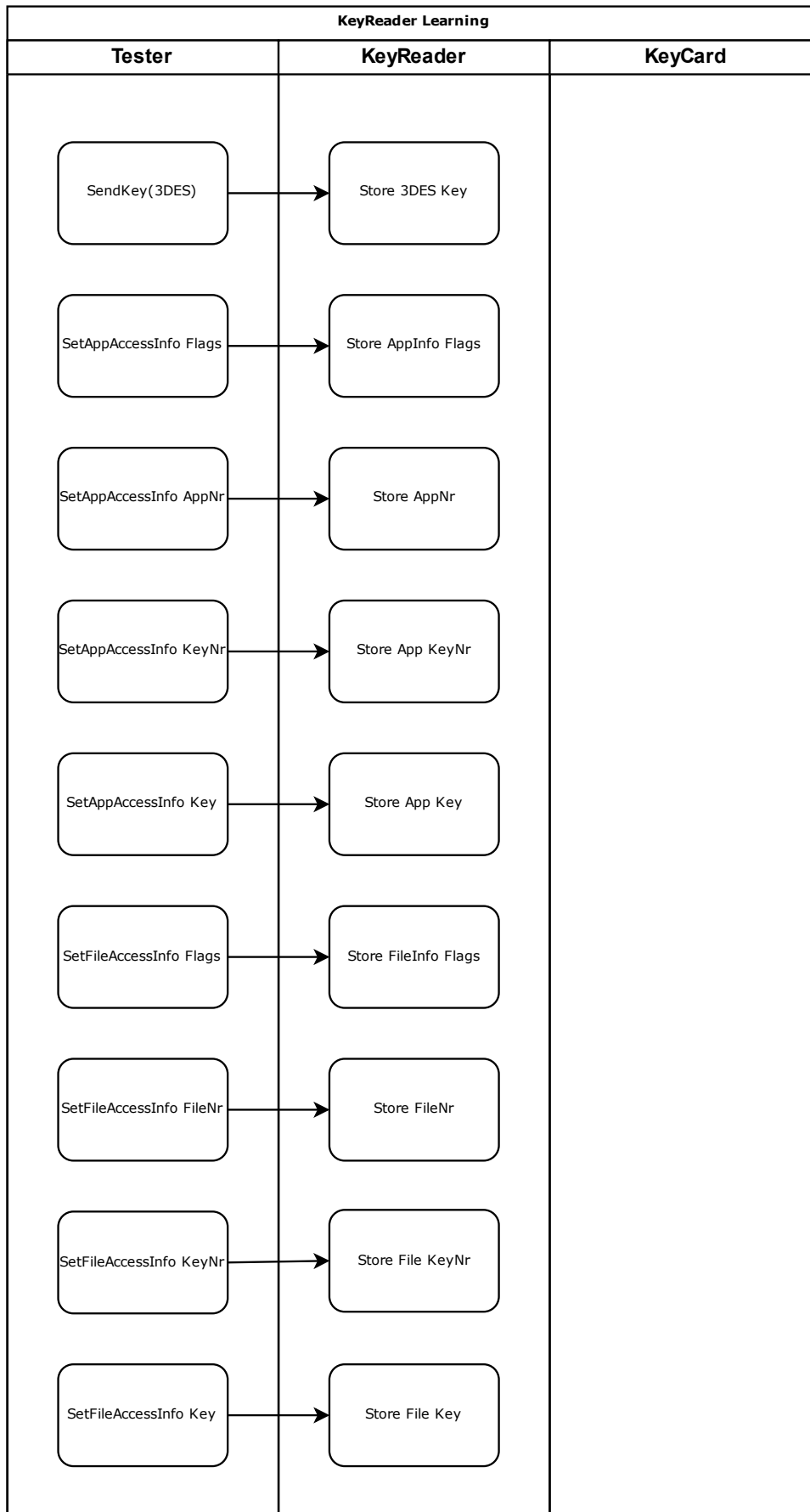
- Send ECU_A_Wakeup (0x0600)
- Wait 2...10 ms
- Send ECU_A_Wakeup (0x0600)
- Wait 400 ms
- Send individual number of RFID-Lock to ECU_A

1.1.2 Initialising the RFID-Lock

Before the RFID device can operate as Lock all access information must be configured by the vehicle computer (or other IT equipment).

The initial KeyReader learning is done one time. This could be at the End of the production line or later on a service partner site. Within this process every Info is written into the KeyReader.

- SetAppAccessInfo (Flags, AppNr, KeyNr, Key, KeySetting)
- SetFileAccessInfo (Flags, FileNr, KeyNr, Key, KeySetting)
- Set3DESKey for encrypted CAN communication



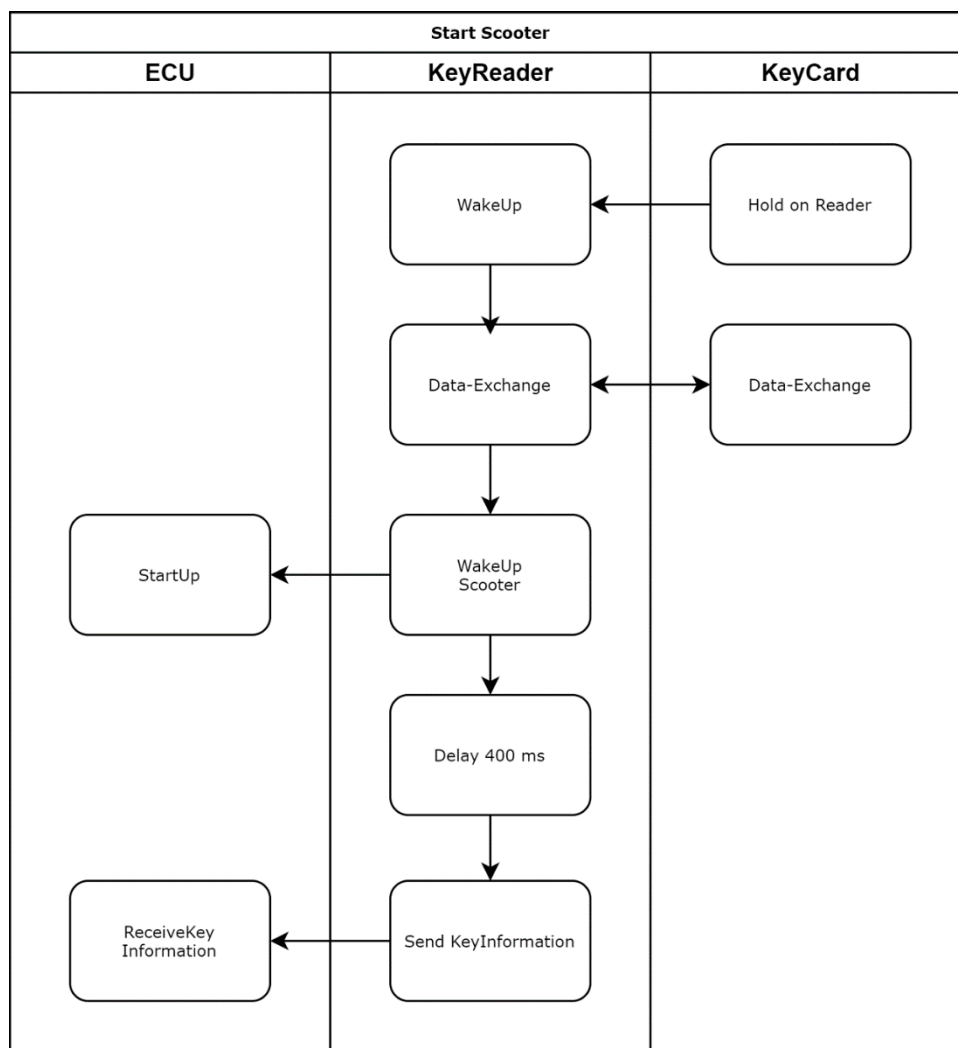
1.1.3 Function Description Key-Operation (Standard Operation Mode)

This will send the 4 Bytes long key number from a file on the DESFire RFID tag using previously initialized information to the ECU A. Before this the ECU A needs to be woken up. The „Send KeyInformation“ is a UDS „Data Write“ to the ECU with the Identifier KeyInformation.

The Process in Detail

- The RFID module detects the RFID keyfob.
- The RFID module wakes up the MCU.
- The MCU will switch the RFID module to standard operation.
- The MCU will send commands to the RFID module to read the key number (4 Bytes) using the previously stored access data from DESFire RFID tag.
- If a key number can be successfully read from the DESFire RFID tag, the MCU will wake up the ECU A
- After 400 ms the MCU will send this number to the vehicle computer.
- The MCU will switch the RFID module to low-power card detection mode.
- The MCU will switch itself to low-power mode.

The Process as Flow Chart



1.1.4 Function Description Key Teach-In Operation

This will write 4 Bytes of the key number into a file on the DESFire RFID tag using previously initialized information. The KeyLearning could be done at every time with a Tester. For this the KeyCard has to be hold to the reader and the tester can then write a KeyID into the Card.

The vehicle computer ECU-A sends a CAN-Bus telegram to the RFID-Lock with the key number (4 Bytes).

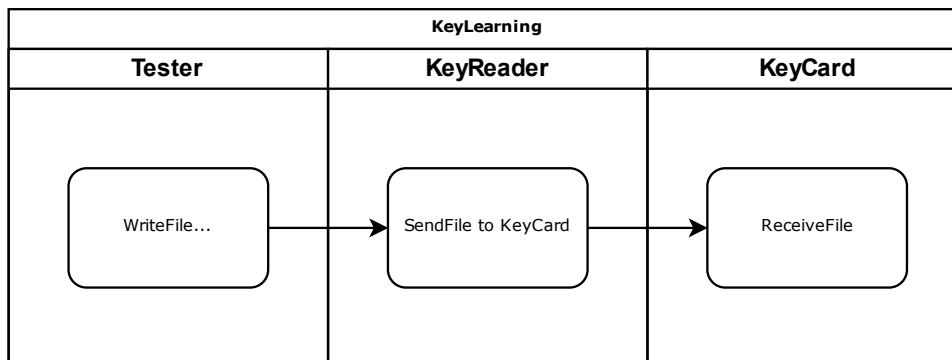
Internally the MCU + RFID will perform several RFID commands to prepare the RFID tag and finally write the 4 Bytes long key number into a file.

The MCU will store the key number onto the DESFire RFID tag:

- PICCActivate
- RATS
- Create Application
- Select Application
- Set password for full file access (file write password)
- Set password for read-only access
- Create File
- Write key number (4 Bytes) into the file

After this, the RFID-Lock shall test if the data is correctly written onto the DESFire RFID tag. If the data is correct, the RFID-Lock will confirm to the vehicle computer.

The Process as Flow Chart (without RFID handling details)



1.1.5 Service Commands

- TransferFirmware2RFIDModule
- TransferFirmware2MCU
- GetVersionInformation from RFID PCB
- GetVersionInformation form CAN-Bus PCB

1.2 Reference Documents

Teach-In Beispiel_x.y_DE.pdf	Example communication to teach in the RFID settings and program RFID-Locks, German
Teach-In Example_x.y_EN.pdf	Example communication to teach in the RFID settings and program RFID-Locks, English

1.3 Glossary

App = Folder on DESFire RFID tag
ECU A = Electronic Control Unit type A, computer of the electric vehicle
MCU = MicroController Unit
PCB = Printed Component Board
PID = Parameter ID
RFID = Radio Frequency Identification
SID = Standard Identifier
TOS = Type of Services
TP = Transport Protocol
UDS = Unified Diagnostic Services

2 CAN-Bus RFID Function

2.1 CAN Settings

CAN Baudrate: 250kBit/s
 CAN Format: CAN 2.0A, Extended frame format

2.2 KeyReader Message

2.2.1 ECU_A_Wakeup

Activate ECU A to activate the roller. Only the message without content is displayed. To wake up ECU A, the message must be sent at least 2 times within 10ms with a minimum interval of 2ms.

Cycle time: Sent only when required.

This message uses the standard frame format and an empty data field. So a single ECU_A_Wakeupt telegram contains this data:

0x0600 = 0b110.0000.0000 (CRC and Stuff-bits not shown)

2.3 UDS Protocol

For communication between KeyReader and control unit, the UDS protocol is used to a reduced extent.

CAN Identifier: 29bit
 Transport protocol for UDS: ISO-TP (ISO 15765-2)

2.3.1 Identifier Structure

29 bit CAN identifier								
28	27	26	25	24	23	22	21-11	10-0
Priority			Extended data page	Data Page	TOS		Source Address	Destination Address

2.3.2 Possible Identifier Values

Not all values in the data fields are possible due to the remaining CAN matrix. Other values other than those specified here must not be used.

Field	Value
Priority	0b000-0b110
Extended data page	0b1
Data Page	0b1
TOS	0b11
Source	0-0x7FF
Destination	0-0x7FF

2.3.3 Addresses

Designation	Address
Tester (PC)	0x00
ECU_A	0x01
KeyReader	0x36

2.3.4 SIDs and PIDs

Service	Request SID	PID	
Data Read	0x22		Read Data by identifier
Data Write	0x2E		Write Data by identifier
Request Download	0x34		Request Download
Request Upload	0x35		Request Upload
Transfer Data	0x36		Transfer Data
Transfer Exit	0x37		Request Transfer Exit
Error	0x7F		Negative Response

To confirm a received message, a message with an SID increased by 0x40 is sent back. The PID remains the same. This message does not contain any data but only serves as positive feedback that the message has been received and processed. If data is requested, then the response message also directly contains the response data (e.g. Data Read).

In case of a negative response, a Negative Response is sent. The negative response message has the SID 0x7F and contains as 1st byte directly following the SID the SID of the faulty message and then a response code.

SID	1st Byte	2nd Byte	Description
0x7F	SID	Response Code	Error Message

2.3.5 Response Codes

Following response codes are possible:

Code	Description
0x10	General Reject
0x11	Service not supported
0x12	Subfunction not supported
0x13	Message Length or Format incorrect – wrong format, message should be repeated in correct format
0x21	Busy, repeat request – the message was dropped and should be repeated
0x22	Conditions not correct
0x31	Request out of Range
0x72	General programming Failure (Standard Error Code used by RFID-Lock)

2.3.6 Data Identifier

Identifier for data fields in the UDS protocol.

In case of ASCII type data, a line break is given by “\n” and the ASCII string is terminated with 0x00.

Commands that belong together are highlighted in the same colour.

Type	Byte 0	Byte 1	Type	Description	Default values***
SetAppAccessInfo	0x60	0x02	3 Bytes 3DES Encrypted	SetAppAccessInfo AppNr	Byte 1, App LSB: 0xEF Byte 2, App: 0xCD Byte 3, App MSB: 0xAB
	0x60	0x03	1 Byte 3DES Encrypted	SetAppAccessInfo KeyNr*, Key for full access to settings and files	Byte 1, Key Number: 0x00
	0x60	0x04	16 Bytes 3DES Encrypted	SetAppAccessInfo Key, this will be written in to the SetSppAccessInfo KeyNr	16 Bytes: 0x766B4665 45394C4B 405F3D3D 3337745A
	0x60	0x11	5 Bytes	SetFileAccessInfo Flags	Byte 1, Comm. Settings: 0x00

			3DES Encrypted	Communication setting (0x03) + Access Rights (0x1000) + File Size (0x0004), example values	Byte 2, Access Rights, LSB: 0x00 Byte 3, Access Rights, MSB: 0x10 Byte 4, Files Size, LSB: 0x10 Byte 5, File Size, MSB: 0x00
	0x60	0x12	1 Byte 3DES Encrypted	SetFileAccessInfo FileNr	Byte 1, File Number: 0x08
	0x60	0x13	1 Bytes 3DES Encrypted	SetFileAccessInfo KeyNr**, Key for read file access	Byte 1, Key Number: 0x01
	0x60	0x14	16 Bytes 3DES Encrypted	SetFileAccessInfo Key, this will be written into the SetFileAccessInfo KeyNr	16 Bytes: 0x5575254A 26533F35 4A586632 4234464D
	0x60	0x21	4 Bytes 3DES Encrypted	WriteFile****, prepares a DESFire tag and stores this number into a file	This command expects 4 Bytes from the ECU
	0x60	0x22	4 Bytes 3DES Encrypted	ReadFile, this command will use this settings and try to read out the contents of the file. The command needs parameters.	Byte 1: start address, LSB Byte 2: start address, MSB Byte 3: data length, LSB Byte 4: data length, MSB
	0x60	0x31	24 Bytes Unencrypted	SendKey (3DES) One Time Write	
	0x60	0x41	Variable length ASCII, Unencrypted	Firmware Version CAN MCU	
	0x60	0x42	Variable length ASCII, Unencrypted	Firmware Version RFID MCU	

* This key shall have full file access rights

** This key shall have read-only file access rights

*** The default values shall allow preparation of a test tag. When a test tag is put to the antenna, the RFID Lock will send the ECU wake-up message.

**** This is the most complex function. It will set up a DESFire RFID Tag to be an RFID Key. This function will create an app, set 2 keys in the app, create a file and store 4 Bytes into the file

3 Firmware Update via CAN-Bus

This chapter describes the update of microcontroller Software within the scooter via the ISO-TP protocol. The ISO-TP protocol is implemented over Controller Area Network (CAN) Bus.

3.1 CAN

The CAN is using a 2.0B "Extended frame format" with 29bit identifier. The initial speed is defined to 250kbit/s but can be configured within the application to another speed.

3.1.1 Identifier structure

The 29bit Identifiers are structured similar to the Identifier in the application. The structure is described in the following table:

29 bit CAN identifier								
28	27	26	25	24	23	22	21-11	10-0
Priority			Extended data page	Data Page	TOS		Source Address	Destination Address

3.1.2 Possible Identifier Values

The possible values for the CAN identifiers are described in the following table:

Field	Value
Priority	0b000-0b110
Extended data page	0b1
Data Page	0b1
TOS	0b11
Source	0-0x7FF
Destination	0-0x7FF

3.1.3 Source and Destination Addresses

Designation	Address
Tester (PC)	0x00
KeyReader	0x36

3.2 Timeout

The bootloader has a timeout of 15min. The timeout time gets reset everytime when a CAN command was received, which is for the controller. If the Timeout happens the controller tries to restart. If there is no application it has to stay in the bootloader mode.

3.3 Protocol

3.3.1 General

The protocol follows a request/response flow.

3.3.2 Data Frame

Request Frame

The data frame within the ISOTP Message for a request is structured as described in the following table:

Byte 0	Byte 1	Byte 2	Byte 3 to x-1	Byte x
Length MSB	Length LSB	Command	Data	CRC

Response Frame

The data frame within the ISOTP Message for a response is structured as described in the following table. The command byte is the same as in the request.

Byte 0	Byte 1	Byte 2	Byte 3 to x-1	Byte x
Length MSB	Length LSB	Command	Data	CRC

3.3.3 CRC Calculation

The CRC is calculated by summing up the whole bytes from length to end of data, then doing a bitwise OR and after that add one.

```
uint16_t sum = 0;
for (uint8_t byte : data)
{
    sum += byte;
}
return ((~sum) + 1);
```

3.4 Commands

There are different commands available.

3.4.1 Readout the Software Version

The command byte for reading out the software versions is 0x00.

Request

For the Request there is no data required.

Response

Within the response there has to be the software version of the bootloader and the application as ASCII format string both separated by a zero termination. E.g. 0x30, 0x2E, 0x32, 0x2E, 0x31, 0x32, 0x33, 0x00, 0x31, 0x2E, 0x31, 0x2E, 0x35, 0x32, 0x31, 0x00 for Bootloader Software Version 0.2.123 and Application Software version 1.1.521. If there is no Firmware in the application the string has to be empty, so a second 0x00 right after the one for the bootloader version.

3.4.2 Erase Flashpage

The command byte for erasing a flashpage is 0x01.

Request

The data in the request contains the flash address.

Data 0	Data 1	Data 2	Data 3
Address MSB	Address	Address	Address LSB

Response

The response is an ASCII 'A' (0x41) for a successfully received and executed command or a 'F' (0x46) for any failure in the erase process.

3.4.3 Write flash

The command byte for writing into the flash is 0x02.

Request

The data contains the first the address where to write the data with four bytes, then the byte count with two bytes and after that a maximum of 2048 data bytes to write.

Data 0	Data 1	Data 2	Data 3	Data 4	Data 5	Data x
Address MSB	Address	Address	Address LSB	Size MSB	Size LSB	Data

Response

The response is an ASCII 'A' (0x41) for a successfully received message or a 'F' (0x46) for any failure in the write process.

3.4.4 Start application

The command byte for starting the application is 0x22.

Request

For the Request there is no data required.

Response

The response is an ASCII 'A' (0x41) message if the application can be started or a 'F' (0x46) for any failure in the starting process, e.g. application CRC check failure.

3.5 Firmware Update Procedure

The firmware update procedure is as follows:

1. Enter bootloader from application
2. Request the software versions*
3. Erase the required flash
4. Write the new application into the flash
5. Start the application

If the application can't be started due to a CRC failure within the application, it has to stay in the bootloader.

* if the software version check shows the version is already up-to-date, the command "start application" can follow without flashing new firmware.

3.6 Application Start cycle

After a normal power reset, the bootloader has to follow this startup procedure:

1. Peripheral Initialisation
2. Bootloader enter check
 - a) Yes: GOTO Bootloader enter
3. Application CRC check
 - a) CRC not valid: GOTO Bootloader enter
4. Start application

4 Electric Requirements

4.1 Power Supply with Overvoltage Protection

Requirement

The supply voltage for the reader is between 7.6 V and 13.8 V.

The supply circuitry shall be able to withstand +60 Vdc and reverse polarity.

In overvoltage situation, there is no need to keep the normal function intact.

The power circuitry in the RFID Lock can switch off in case of overvoltage or reverse polarity.

4.2 Current Consumption

Requirement

Sleep: During sleep mode the current consumption has to be « 1 mA.

Active: During active mode the current consumption can be up to 120 mA.

RFID Module: < 50 mA

STM32L: 62 µA/MHz

Low-Power Mode

RFID Module: unknown

STM32L: 3 µA

4.3 CAN-Bus with Overvoltage Protection

The CAN-Bus shall be protected to withstand +60 Vdc.

In overvoltage situation, there is no need to keep the normal function intact.

4.4 Connector

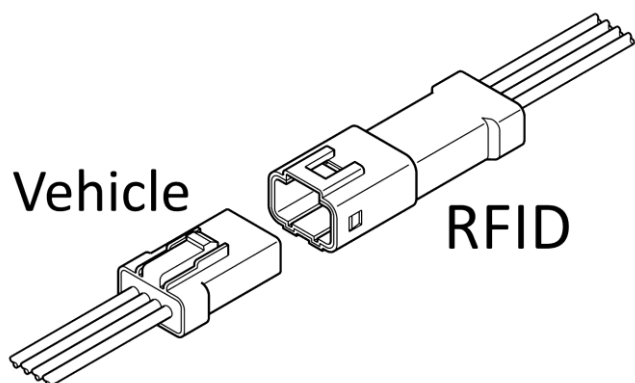
4.4.1 Connector Information

Connector series	JWPF by JST
Connector housing, 4 pin	04T-JWPF-VSLE-S
Crimp contacts, 4 pcs. per housing	SWPT-001T-P025
Wire gauge	0.33 mm ² \triangleq AWG22
Data Sheet	jwpf-wtw.pdf

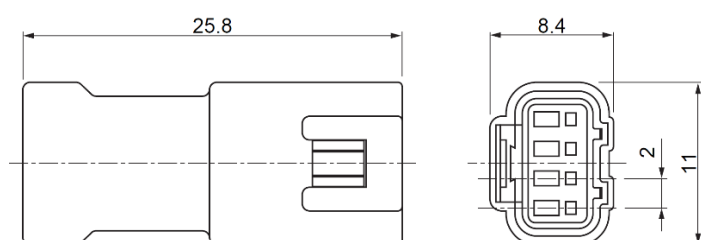
4.4.2 Connector Pinout

Pin	Colour	Function
1	Yellow	12V
2	Orange	CAN-High
3	Violet	CAN-Low
4	Blue	GND

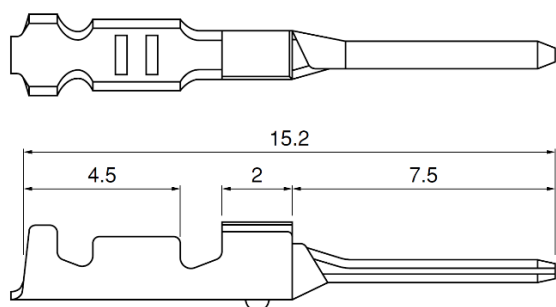
4.4.3 Connector Components



4.4.4 Dimensions Connector Housing



4.4.5 Dimensions Crimp Contact



4.5 SWD connector for Firmware update of the CAN-Bus MCU

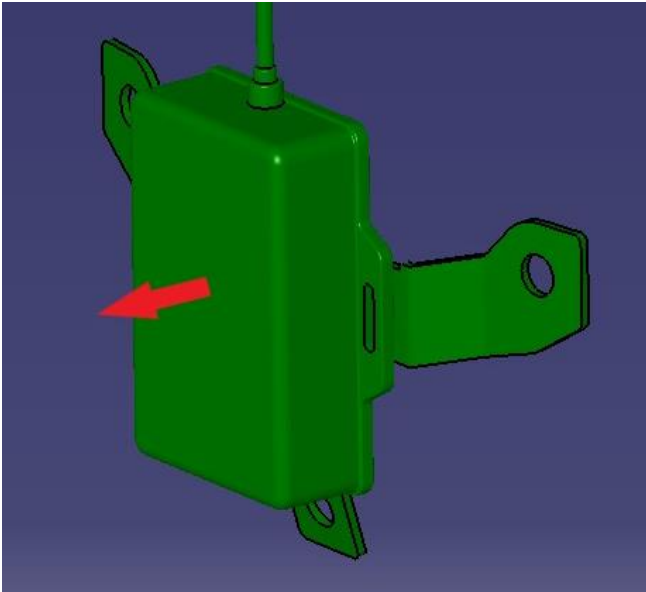


4.6 Hardware Information

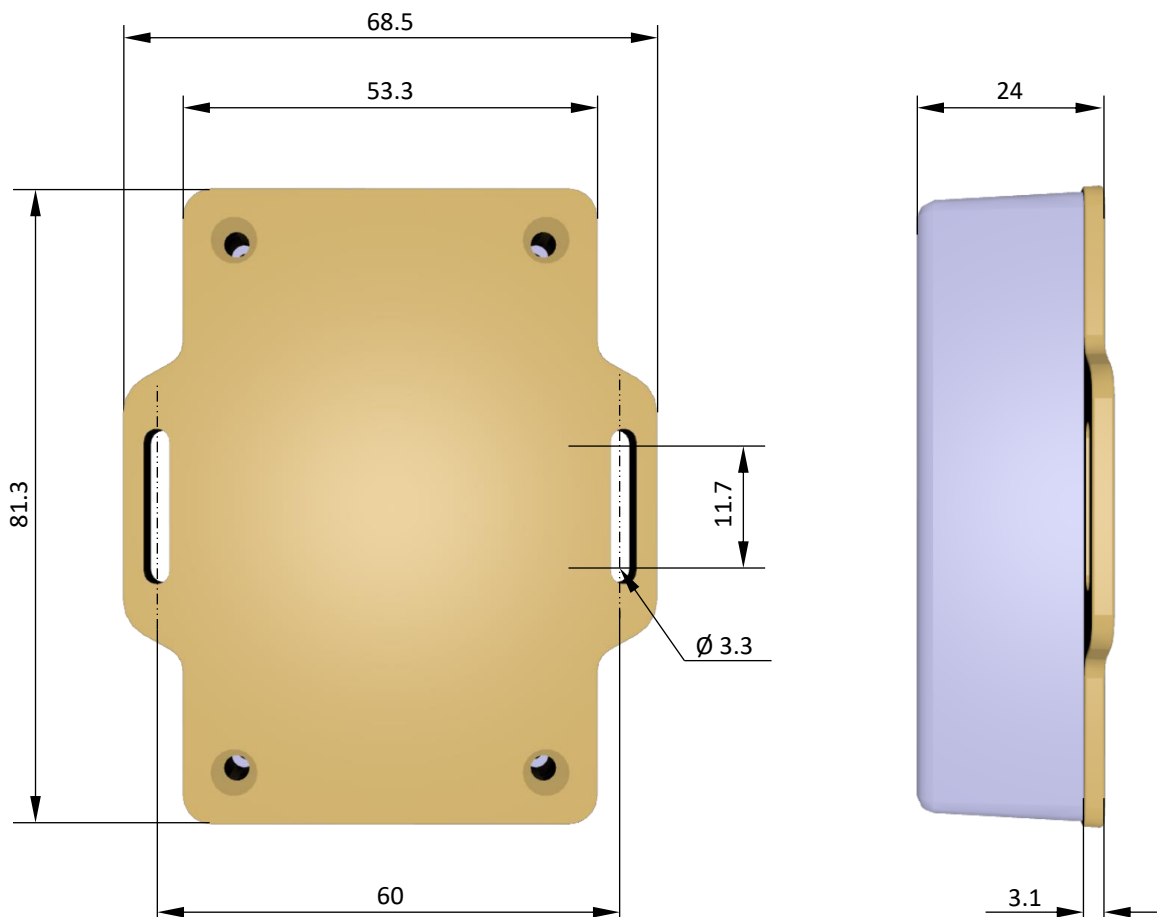
CAN-Bus Interface IC:	Texas Instruments SN65HVD231DR
CAN-Bus MCU IC:	STMicroelectronics STM32F103C8T6

5 Mechanics

5.1 Antenna Read Direction



5.2 Dimensional Drawing



6 Technical Data

Electrical Specifications	
Power Supply	7.6 ... 13.8 V
Power Consumption	< 120 mA, standby current < 1 mA (low power mode)
Operating Frequency	13.56 MHz
Baudrate CAN-Bus	250 kbit/s, Order code: R-PROF-DES-LOCK-CAN-KE 500 kbit/s, Order code: R-PROF-DES-LOCK-CAN-KE-500
Antenna	Internal
Reader IC	CL 663
RF TX Speed	up to 848 kBd
Interfaces	CAN-Bus with custom-specific communication protocol
Environmental Conditions	
Operating Temperature	-10 °C ... +50 °C
Storage Temperature	-20 °C ... +60 °C
Protection Class	IP64 or more (potted electronics)
Humidity	up to 95 %, non condensing
MTBF	200'000 h
Applicable Standards	
EMC	EN 301489-1:2012-04 (v1.9.21) EN 301489-3:2013-12 (V1.6.1)
Radio Regulation	EN 300330-1:2015-08 (V1.8.1) EN 300330-2:2015-08 (V1.6.1)
Safety	EN 50581:2012 (valid till 2024-07-07) EN 63000:2018
RoHS 2	EC Guideline 2011/65/EU, amendment 2015/863
REACH	EU Guideline 1907/2006, updated by 2018/2005/EU
Certificates	FCC, CE

Other functions and details to be continued and upgraded.

7 Revision History

Date	Version	Description
2023-02-22	1.0	First edition of User's Guide
2023-02-28	1.1	Order Codes changed, Reference Documents added
2023-04-11	1.2	Wire Colours added