User Guide
AC-01 V2
USB, RS-232 and OEM

Revision 3

1. Products covered by this guide

This guide contains informations of the following products:

Reader	Code	Details
LEITOR UHF AC-01 V2 USB CABO 2M BR	100.510	*With enclosure.* *With an internal ceramic antenna.*
LEITOR UHF AC-01 V2 RS232 CABO 2M BR	100.549	
LEITOR UHF AC-01 V2 OEM USB	100.511	*Without enclosure.* *With a ceramic antenna.*
LEITOR UHF AC-01 V2 OEM RS232	100.550	

1.1. Approval Note

National Telecommunications Agency (ANATEL):

The Acupad-50 and AcuPad-50 Mux readers, was tested and approved in accordance with the Regulation for Certification and Homologation of Telecommunications Products, approved by Anatel Resolution No. 242 of November 30, 2000.
Types: Radio Frequency Identification Systems - Category II.
Service/Application: Restricted Radiation Radiocommunication.
"This equipment is not entitled to protection against harmful interference and may not cause interference to properly authorized systems."

Federal Communication Commission Interference Statement (FCC)

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) This device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one of the following measures:
- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

⚠️ **FCC Caution:** Any changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate this equipment.

> ⚠️ **MPE Warning:** Radiation Exposure Statement – This equipment complies with FCC radiation exposure limits set forth for an uncontrolled environment. This equipment should be installed and operated with minimum distance 21cm between the radiator & your body. Contains FCCID: QV5MERCURY6EN

Authorized Antenna

General Characteristics of the Integrated Antenna of AC-01 V2 and AC-01 V2 OEM.

Vendor	ABRACON LLC
Model	APAE915R2540ABDB1-T
Frequency Range (MHz)	915-928 MHz
Circular Peak Gain (dBic)	4.5 dBic
Linear Peak Gain (dBi)	1.5 dBi
Polarization	Circular (RHCP)
Type	Ceramic Patch Antenna

1.2 Legal notice

While every effort has been made to ensure that this document and the information contained therein is correct, ACURA Technologies and any other parties involved in the creation of this document declare that it is provided "as is", without any express or implied warranty, including, but not limited to, any warranties that use of the information contained herein does not infringe any right, legitimacy or fitness for purpose, and therefore waives any liability, direct or indirect, for loss or damage related to the use of this document.
The information contained in this document is subject to change without notice.

1.3 About Acura

Since the end of the 1990s, ACURA has been the pioneer in the radio frequency identification (RFID) market in Brazil and Latin America, and has successfully explored, since the late 1990s, its large-scale adoption in the most diverse sectors of the economy, from mining to steel, agriculture to food processing, from logistics to retail, from transportation to the distribution chain, access control to asset management. Promoter of new technologies, innovative, agile, and focused on the feasibility of cutting-edge projects.

Technological Development and Commercial Office
Wall Street Business
Av. Antartico, 381 - Jardim do Mar, São Bernardo do Campo - SP,
09726-150 (11)3028-4600

2. Introduction

2.1. About this Guide

This guide provides detailed instructions for installing, connecting, configuring, operating, programming and troubleshooting the AC-01 V2 reader. This guide is divided in three parts, Part 1 with physical (mechanical) installation informations, Part 2 which contains detailed informations about Autonomous operation, configuration and troubleshooting, and Part 3 with informations about the Transparent operation as well with programming development information.

2.2. Document conventions

Throughout this document some important informations should be highlighted using the following notes:

> ⚠️ **WARNING:** This type of warning shows important informations that requires the user's attention.

> **NOTE:**   Important information and tips regarding the subject.

2.3. Definition terms

Terms and definitions specific to this document that supersede any normative references are as follows:

Tag - Any RFID tag that complies with EPCglobal UHF Class 1 Generation 2 standard.
EPC id - Refers to the tag's id number, in the EPC memory bank.
GS1 standard - Refers to EPC Tag Data Standard version 1.11 which is the GS1 tag codification standard.
USB - Abbreviation of Universal Serial Bus, is an industry standard that establishes specifications for cables, connectors and protocols for connection, communication and power supply between personal computers and their peripheral devices
CDC - Short for Communication Device Class, used in USB communication.
ACM - Short for Abstract Control Model which is a type of CDC.
HID - Abbreviation of Human Interface Device, is a device class of USB.
ASCII - Abbreviated from American Standard Code for Information Interchange, is a character encoding standard for electronic communication.
<CR> - Carriage Return, a control character, in this document represent the [ENTER] key.

2.4. Target audience

This document is intended for helping the professional in set up and install the AC-01 V2 USB. Before attempting to install, configure, and operate this product, you should be familiar with the following terms:

- Device communication parameters including USB communications.
- Basic knowledge about driver setting for Windows OS.
- Basic knowledge about RF antenna radiation and positioning.
- Basic knowledge about EPCglobal UHF Class 1 Generation 2 standard protocol.

3. Introduction to equipment

The AC-01 V2 is a desktop small factory RFID UHF (Radio Frequency Identification) (Ultra High Frequency) reader with great use flexibility, this device can be used in several applications, such as tag registration in Access Control deploy, such as checkout application in point of sales and many others. Following are the main features of the reader:

- Can work in two different ways, Autonomous and Transparent:
  - **Autonomous mode:**
    - Automatically performs tag read operations continuously;
    - Send the tag reading results in the following interfaces:
      - Via keyboard emulation (native);
      - Via serial communication;
      - Via keyboard emulation + serial communication;
    - Several configuration parameters via serial terminal:
      - Read power;
      - RF times;
      - EPC Gen2 parameters (tag>>reader) (reader>>tag);
      - Device information;
      - Tag reading operation;
      - Tag RSSI value reading filter;
    - Decodes GS1 SGTIN-96 and SGTIN-198 standard tags for checkout applications, sends in the following formats:
      - gs1string, e.g., "(01)03608449920322(21)1540341";
      - gs1epcuri, e.g., "urn:epc:tag:sgtin-96:1.360844.0992032.1540341";
      - gs1epcpureuri, e.g., "urn:epc:id:sgtin:360844.0992032.1540341";
      - gs1gtin13, e.g., "3608449920322";
      - gs1sgtin13, e.g., "3608449920322 1540341";
    - Decodes tags in the Acura Autoid System standard, for tag registration in Vehicle Identification Application (Access Control);
      - autoid + autoidcs
      - wiegand26, e.g., "172 13259";
      - magstripe, e.g., "00001034695627";
    - Decodes tags in ASCII pattern (Maintenance or General application) ;
    - Option to not decode and send the tag's EPC hexadecimal "pure" value;
    - In addition to reading and decoding the EPC tag, it is possible to read any memory bank from the tag;
  - **Transparent mode:**
    - The reader should be controlled by a Software that uses MercuryAPI, a complete SDK for reading and writing tags in the following languages:
      - C, .Net (C#) e Java;
- Compatible with Windows 10, Mac and Linux without driver, for Windows 7 and 8 the driver is provided for installation;
- It has reduced dimensions, not impacting on the aesthetics of the installation site;
- Wide reading power range: 0 to 15dBm in Autonomous mode and 0 to 27dBm in Transparent mode (using a Y extender cable for an additional power input).

4. Revision tables

4.1. Document revision table

Revision	Date mm/yyyy	Description
3	07/2019	⬧　Added information about the OEM variants.
2	06/2019	⬧　Added information about the model AC-01M V2 which can read tags with SJ5511 protocol (Brazilian toll system).
1	10/2018	⬧　First issue.

4.2. Firmware revision table

Revision	Date mm/yyyy	Description
1.1.2	07/2019	• Added support to "Micro" rfid module; • Added read commands for Brazilian's toll tags; • Minor fixes and improved reader stability.
1.0.6	03/2019	• FIxed bug related to internal communication that generated instability if the RFON time was set to more than 500ms.
1.0.2	10/2018	• First issue.

4.3. Hardware revision table

Revision	Date mm/yyyy	Description
V1	08/2018	• First release.

5. Installation notes and important warnings

| ⚠️ WARNING: | The maximum reader read range for passive tags is 20cm (0.65ft) for models with ceramic internal antenna, for OEM model the distance will depend on the type of external antenna and the length of the RF cable used and may reach a few meters away. The effective tag read range may vary depending on:
• Material tag is installed on.
• Tag positioning.
• Reader's direction according to the tag.
• Electromagnetic interference from other devices near the reader |

| NOTE: | We recommend to run some tests in the installation site before the final position, following the recommendations and the customer needs. |

| ⚠️ WARNING: | There is a variation of read distance of a tag in vertical and horizontal orientation, due to the internal ceramic antenna. |

Part 1 - Overview and Installation

Equipment overview

6. Equipament overview

6.1. Hardware - AC-01 V2 USB / AC-01 M V2 USB

Fixed cable

USB cable 2m

RFID Antenna

AC-01
Leitor UHF USB

Usb A

Red LED   Green LED

13

[mm]

AC-01
Leitor UHF USB

89

20   4.5

Rubber foot

57mm
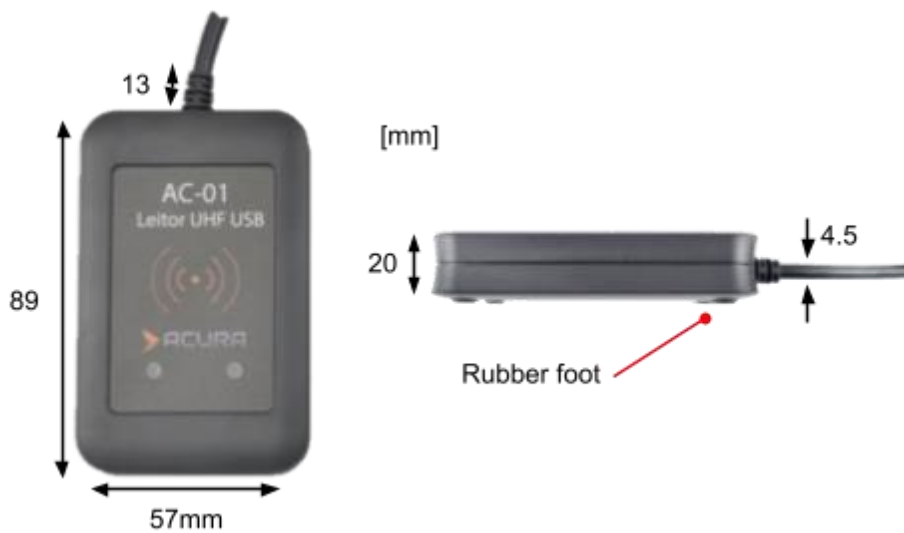
6.2. Hardware - AC-01 V2 RS-232

Power adapter 5V/3A

RS-232 cable 2m

P4 connector

AC-01
Leitor UHF USB

DB9 female
connector
RS-232
2 - TXD
3 - RXD
5 - GND

Female P4
connector

6.4 Technical specifications - reader

Transponder protocols	
Protocol	EPCGlobal Gen2 (ISO 18000-6C). EPCGlobal Gen2 V2 (ISO 18000-63) (transparent mode).
RF interface	
RF power output	Autonomous mode: 0 to 15 dBm. Transparent mode: 0 to 27 dBm (above 15 dBm, a Y extension cable is required to connect one more USB port for power).
Compliance	ANATEL (BR) 915 - 928 MHz. EMC FCC 47 CFR, Part 15.
Mode / Modulation / RF Encoding	Frequency Hopping / PR-ASK / M2, M4, M8.
Backscatter Link Frequency (BLF)	250KHz.
Performance	
Max tag read distance[1]	20cm.
Control / Data interface	
Connectors	**Power and communication:** USB connector type A male.
Data communication	**USB HID keyboard:** The reader is recognized by the Operating System (Windows, Mac and Linux) as a keyboard type HID device. **USB Serial CDC:[2]** When the reader is plugged into USB port, a Serial port is created automatically (Windows 10, Mac, Linux).
Programming / Configuration	**Autonomous mode:** Configuration via serial communication with ASCII commands via terminal (Putty, Minicom, Screen, etc). It does not require the use of API / SDK. Tag reading only. **Transparent mode:** The reader starts to respond to a Software that uses the Mercury API and has features of reading and writing tags.
Proteção	
Proteção contra ESD	Passive high pass filter (40 dB attenuation in ESD discharge range).

---

[1] Reading distance may vary depending on the tag model, mode of use and the environment.

[2] Windows 7 and 8 requires driver installation.

Power	
DC power requirement	DC voltage: 5VCC +/- 1%. (Host's USB port). Max. ripple: 200mVpp @20MHz.
DC power consumption	Max 2,5W in Autonomous mode. Max 6W in Transparent mode.

Mechanical / Environmental	
Water and Dust Tightness	Indoor usage, no weather protection.
Dimension	89x57x20mm
Operational temperature	-10°C a + 65°C
Storage temperature	-10°C a + 70°C
Humidity	95%
Mount kit	Desktop reader, no mount kit.

6.5 Technical specifications - integrated antenna

Electrical characteristics	
Frequency range	915 - 928 MHz
Circular Peak Gain	4.5 dBic
Linear Peak Gain	1.5 dBi
VSWR	1,5:1 (max.)
Polarization	Circular (RHCP)

7. Installation

7.1. Turning on the reader

Once plugged to a live USB port the reader will turn on in 4s and the following feedback pattern can be showed:

	Beep | Green LED blinks fast and Red LED on: Reader running in **Autonomous mode**, trying to read tags.
	Beep | Green LED and Red LED Blink Fast Simultaneously: Reader powered but not initialized USB communication with the Operating System. Reader is not running until communication with the Operating System occurs.
	Beep | Green LED flashes slowly and Red LED off: Reader running in **Transparent mode**.

> **NOTE:** In Transparent mode, the Red LED and the Beep are controlled by the User Software.

On computers with Windows Operating System prior to 10, you will need to install a driver to create the COMxx serial communication port.
For Linux, Android and Mac operating systems, the driver is not required.
When the reader is powered on, the following devices will be created in the Operating System:
- HID class keyboard;
- Serial port, (COM&LPT on Windows), (CDC on Linux, Android and Mac).

The HID class keyboard will be used by the reader for keyboard emulation while sending the tag reading result in Autonomous mode and the Serial port will be used for communication and configuration in both modes Autonomous and Transparent.

7.1.1. Identifying the Serial port (Windows)

After connecting the reader, go to "Control Panel" or with keys ('Windows' + 'x') open "Device Manager". In "Ports (COM & LPT)", a virtual serial communication port should appear 4s after the time the reader is connected, plus a new HID Keyboard type device in "Keyboards".
The image below shows that the reader has been recognized by the Operating System and is ready for operation.
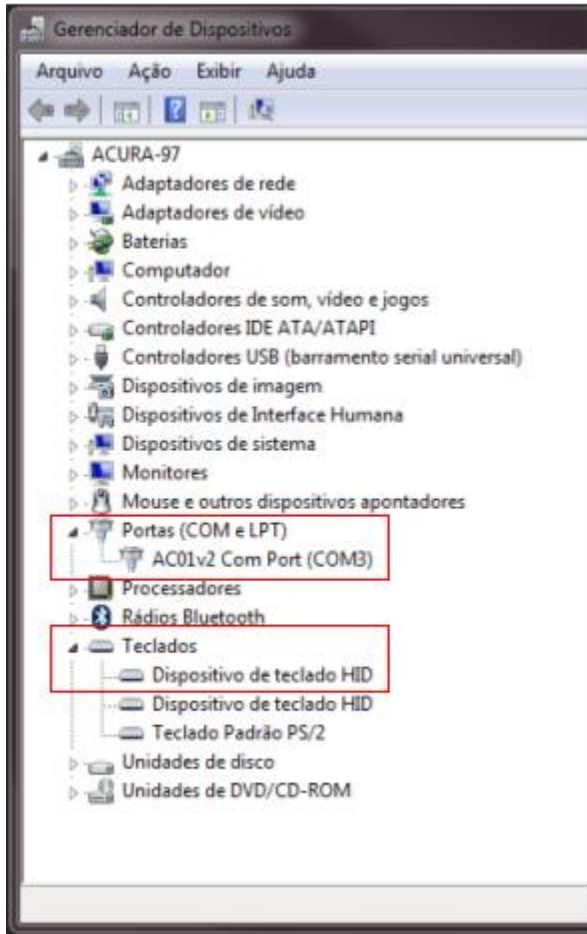
If Windows does not recognize the reader, as in the image below of the Device Manager, the reader's driver must be installed following the following steps.

7.1.1.1. Installing the driver for Windows

The Driver for Windows download must be done through the link:
https://drive.google.com/file/d/1PJBSm4KYy0w-L7tRK-BRwGBatqpwLF3r/view?usp=sharing

7.1.1.1.1. Step 1: Device detection

1. Connect the reader to the computer's USB port;
2. In "Control Panel" or keys ('Windows' + 'x') open the "Device Manager";
3. In "Other Devices" the reader connected to the computer should appear as follows
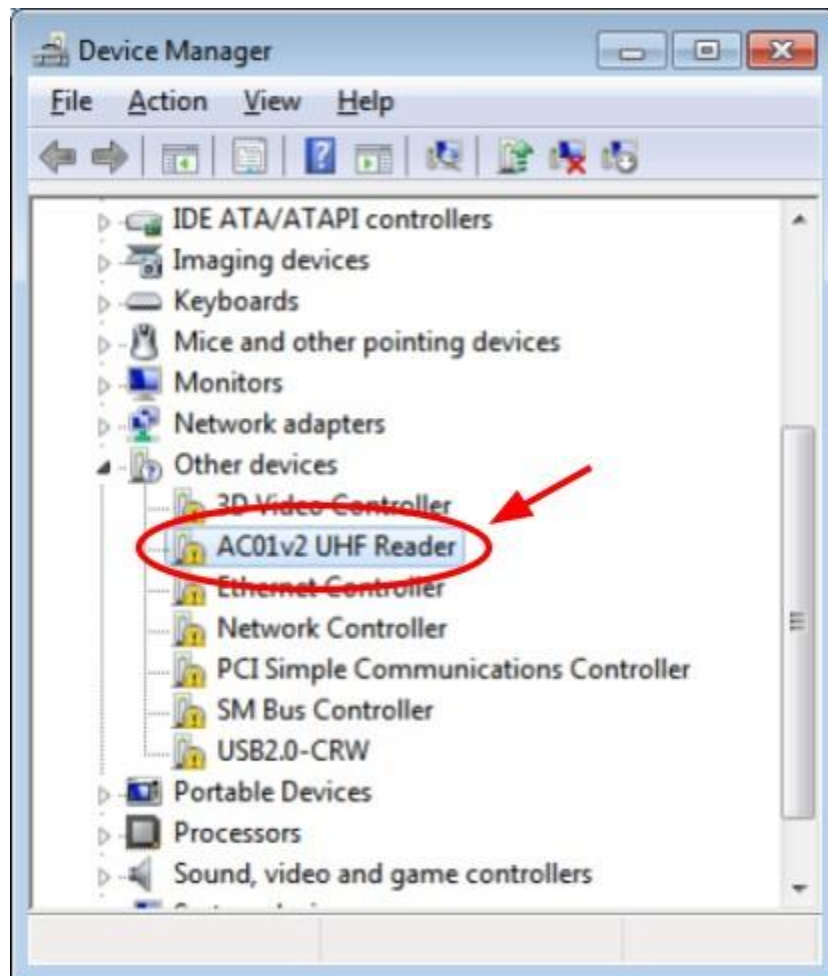
> **NOTE:** Driver installation is not needed for Windows 10 or above, Linux and Mac OS.

7.1.1.1.2. Step 2: Load the driver

1. With the right mouse button, go to "**Update driver...**"
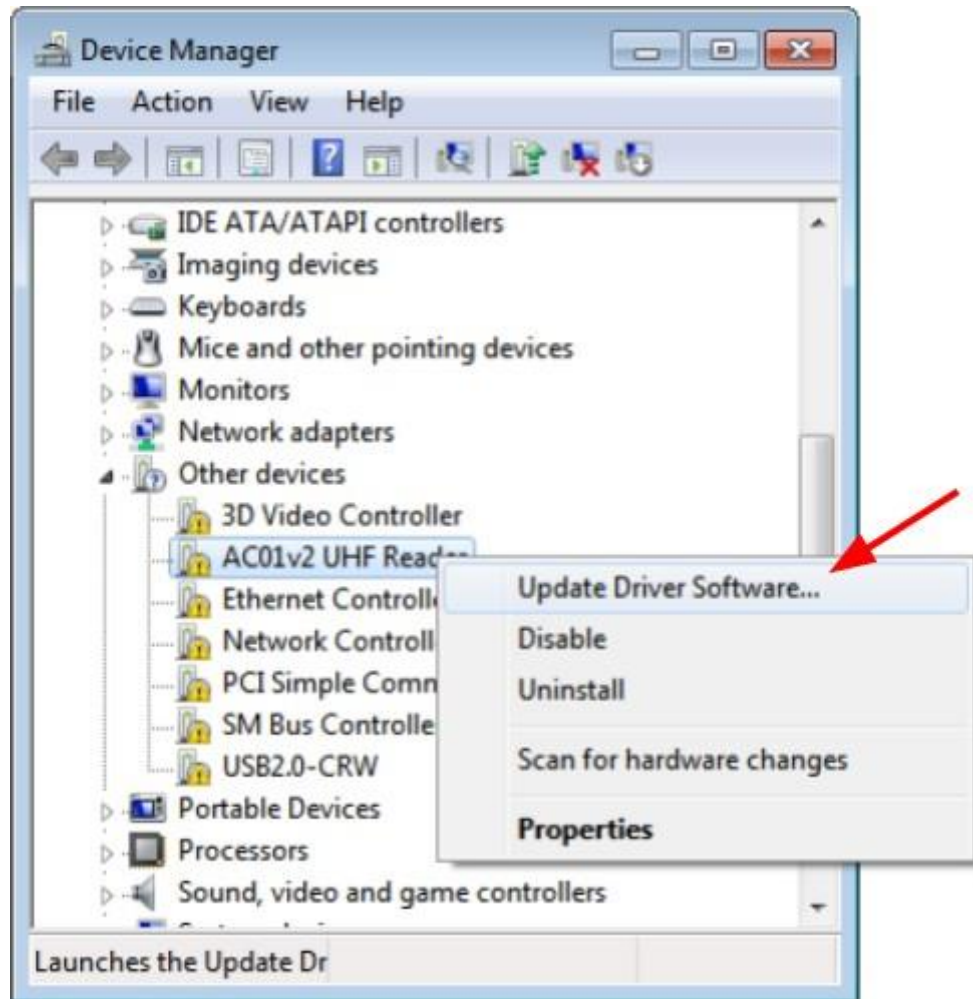
2. Next, "*Browse my computer for driver software*":
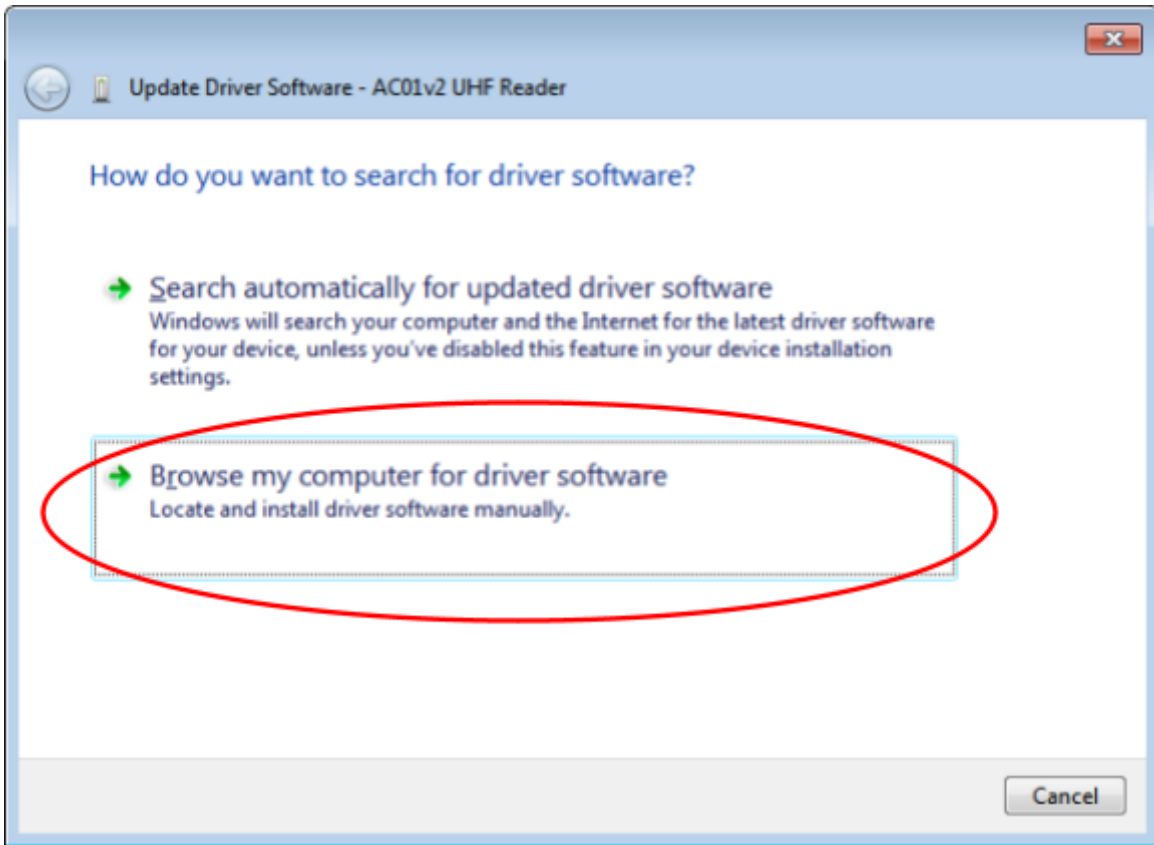
3. Then select "*Let me pick from a list of device drivers on my computer*":

4. In "Common hardware types" select "**Ports (COM e LPT)**" than click "**Next**":
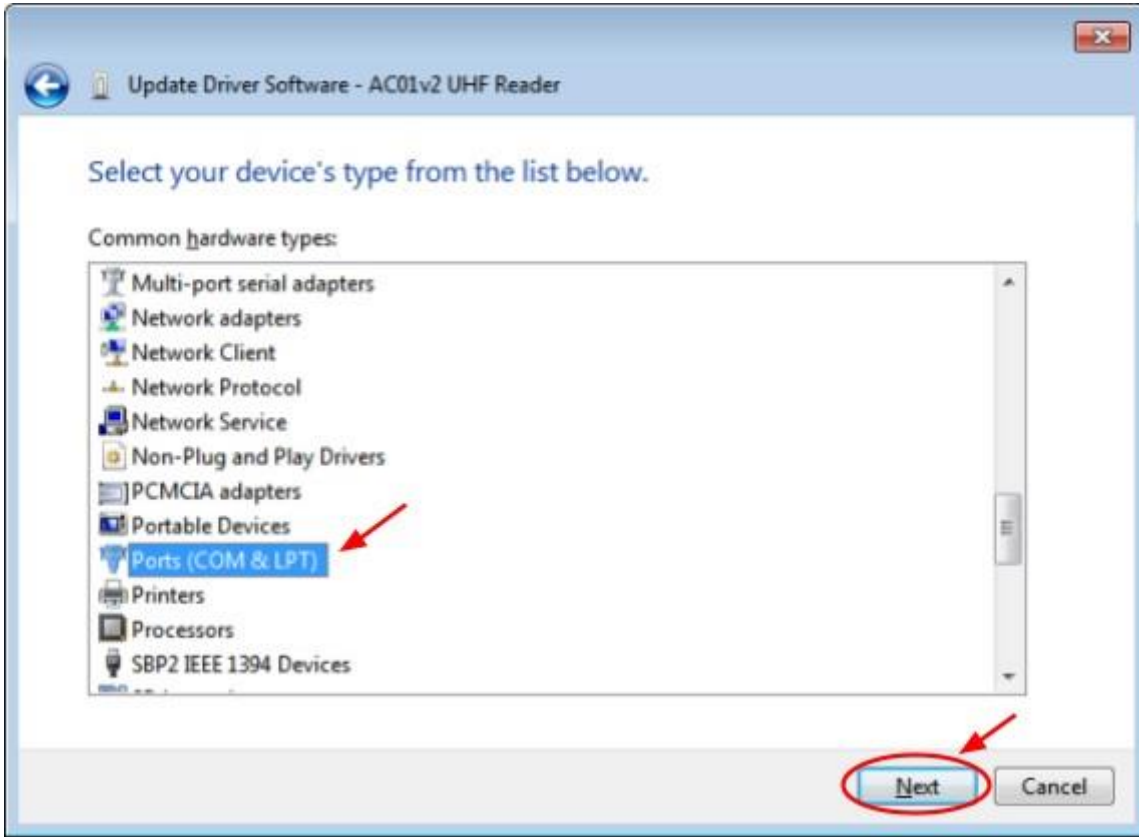
5. Click "**Have Disk…**";
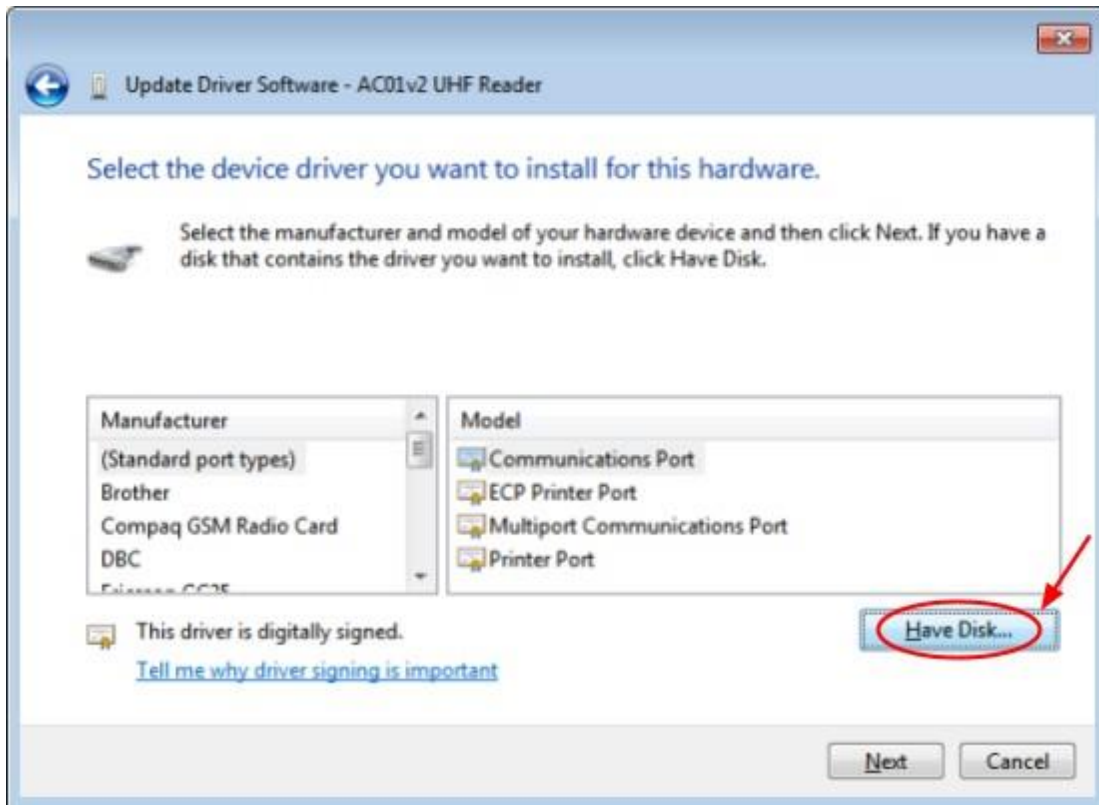
6. Select the "acura_ac01v2.inf" file and click "**OK**". Choose the model and click "**Next**";
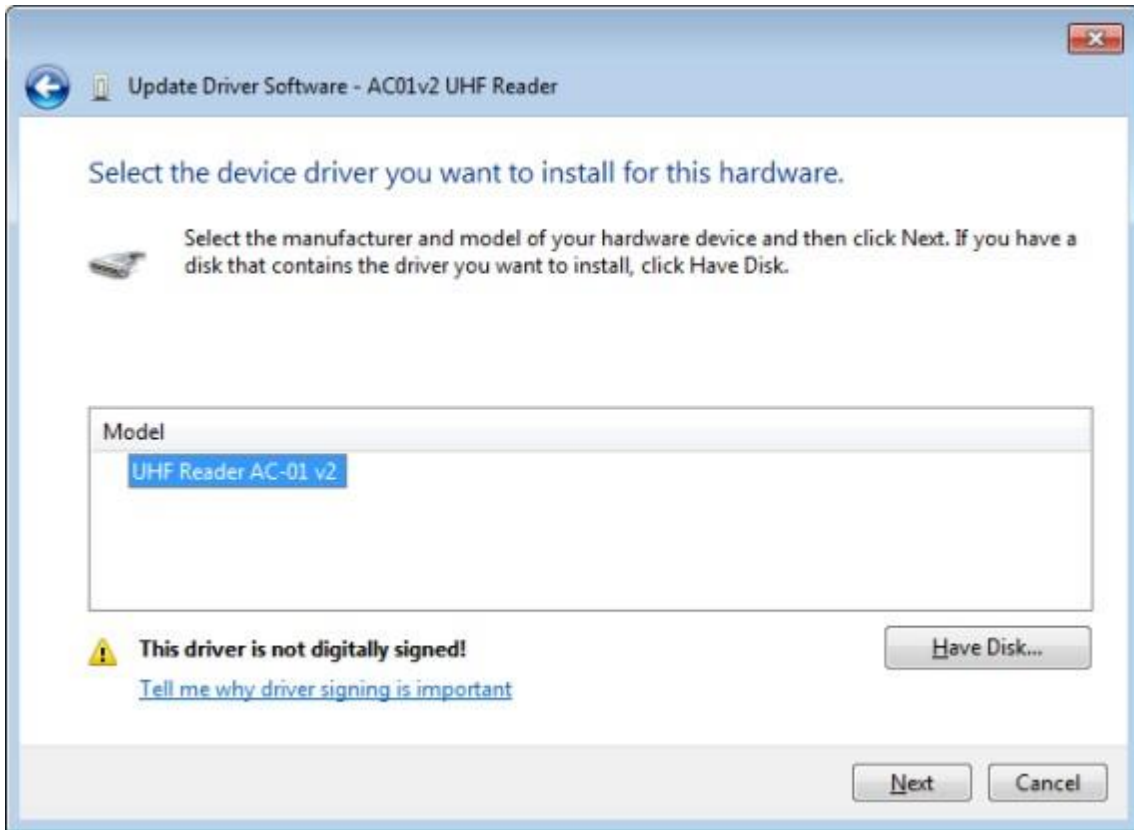
7. A driver update warning should appear. To proceed, click "Yes" to continue installing the driver;

8. A Windows security alert saying "Windows can't verify the publisher of this driver software" appears, click in "**Install this driver software anyway**";
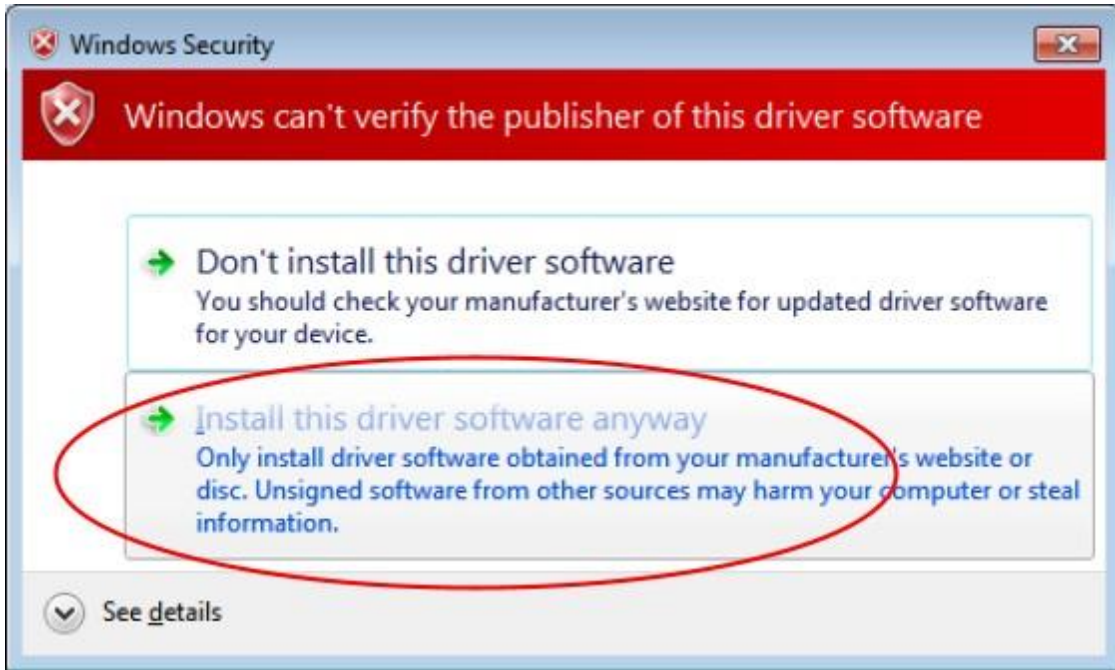
9. After this procedure, wait for the installation;
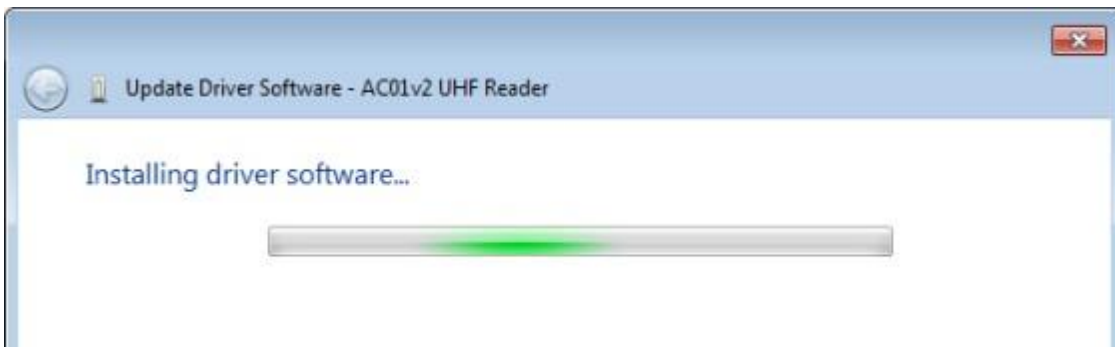
10. Once installed, the successful update message appears.

7.1.2. Identifying the Serial port (Linux)

To identify the name of the serial port created, right after connecting the reader to USB, type in Terminal:

$ dmesg | tail | grep tty
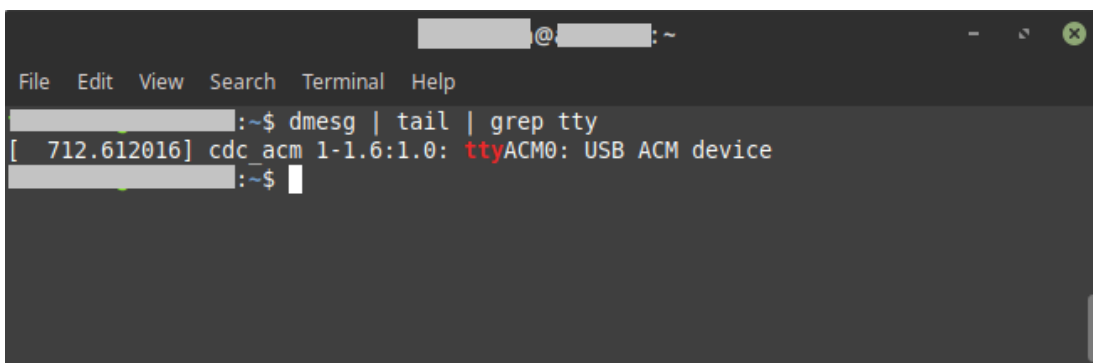
In the above example, the serial communication with the reader will be via **/dev/ttyACM0**.

7.1.3. Identifying the Serial port (Mac OS)

To identify the name of the serial port created, right after connecting the reader to USB, type in Terminal:

$ ls /dev/tty.usb*
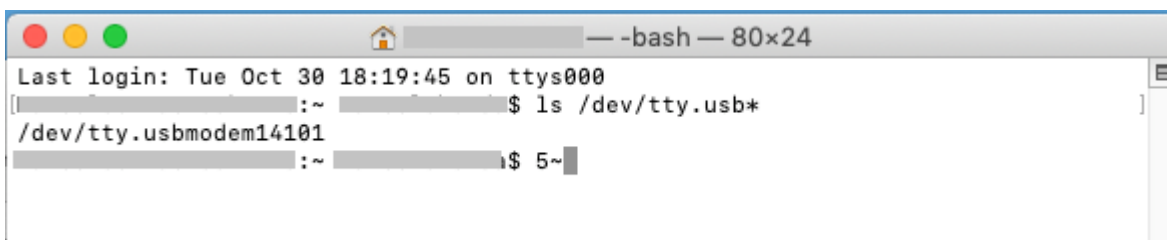
In the above example, the serial communication with the reader will be via **/dev/tty.usbmodem14101**.

Part 2 - Autonomous mode

Reader in Autonomous mode

Configuring the reader

Usage examples

8. Reader in Autonomous mode

In Autonomous mode, the reader constantly tries to read a tag and when that happens, the information is sent through the configured communication interface.

Visual feedback when in Autonomous mode:

Green LED blinks fast and Red LED on
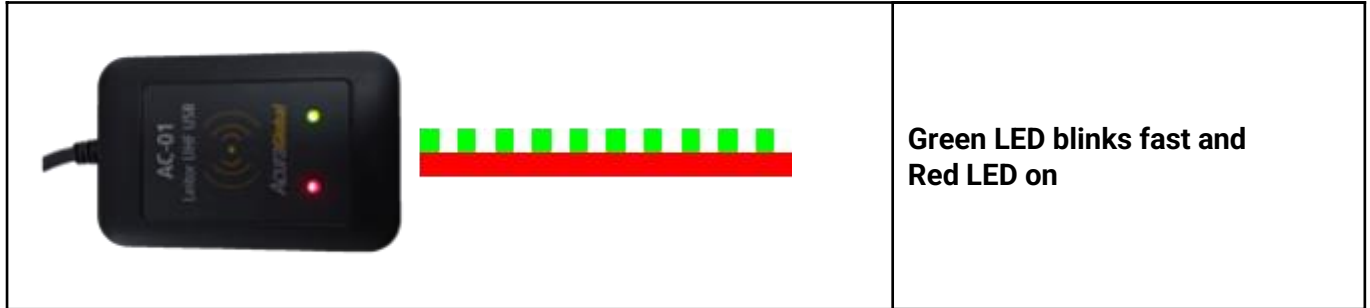
General diagram:
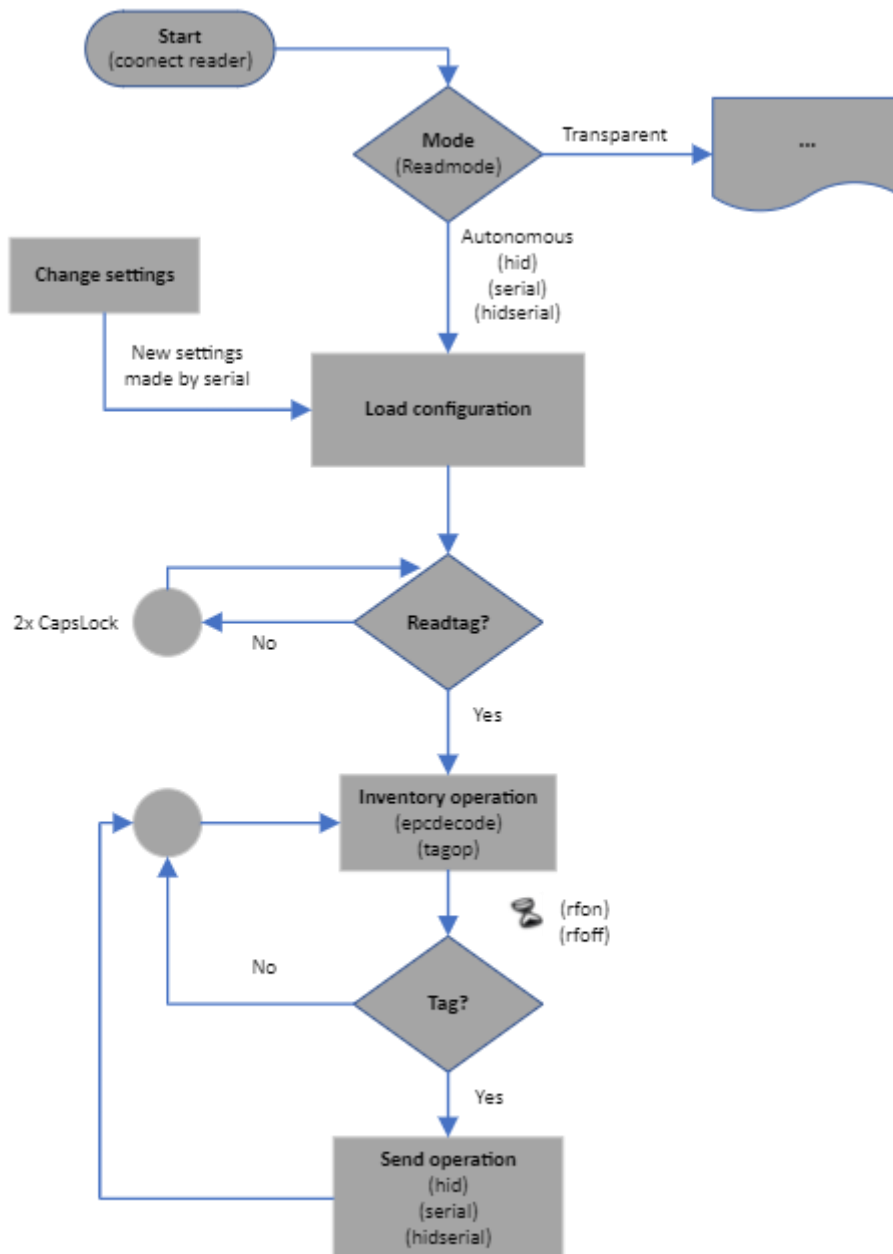
The reader with the most basic configuration reads only the tag's EPC id in hexadecimal format. The tag's EPC id will always be read and reported by the reader and its decoding can be changed. Besides to reading the EPC id it is possible to read together any memory bank of the EPC Gen2 UHF tag, as well as the reading count and its RSSI signal level. All this information is sent by the reader in a formatted String as explained in the next chapter.

8.1. Read result format

In Autonomous mode, the reader sends a formatted String with the read results of the tag in the configured communication interface.
Below is the format of String:
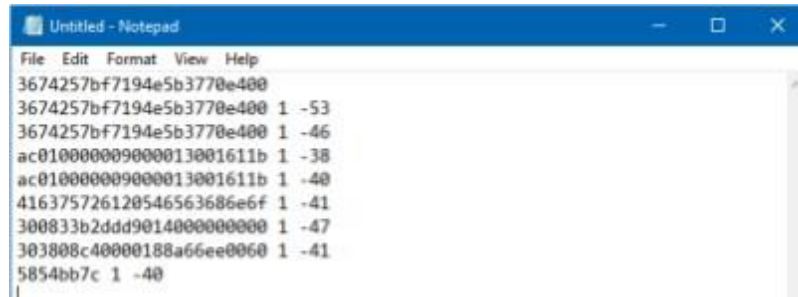
<epcdecode><separator><tagop><separator><readcount><separator><rssi><endofline>
<epcdecode> Read result of EPC id with the decode specified by the command **epcdecode**. *Note: Always reported by the reader in Autonomous mode.*
<separator> Character separator specified by the command **separator**. *Note: It is only reported by the reader if there are more fields than <epcdecode> to report.*
<tagop> Read result of operation specified by command **tagop**. *Nota: Optional.*
<readcount> Read count of same tag in an inventory period. The option to show or not the value is done by the command **reportreadcount**. *Nota: Optional.*
<rssi> Value of RF signal level in dBm in negative, of tag response. The option to show or not the value is done by the command **reportrssi**. *Nota: Optional.*
<endofline> End of line type configured by the command **endofline**. *Note: Reported by the reader in Autonomous mode if set to other than none.*

> **WARNING:** **The result string size may vary depending on the tags read, so if it is a User Software that will receive this string, always process it using the <separator> and <endofline> fields to separate the desired fields.**

Reading results examples:

8.2. Configuring the reader

When the reader is in Autonomous mode, the following settings can be made:
- Communication type;
- Read power;
- Read time;
- Idle time;
- Tag's EPC id decoding type;
- Tag's memory bank read operation;
- Parameters of the EPC Gen2 protocol:
  - BLF;
  - Tari;
  - Encoding;
  - Session;
  - Target;
  - Q (Anti Collision);
  - Access password;
- Character that separates the fields in the result string;
- End of line type of result string;
- Typing speed in keyboard emulation;
- RSSI filter;
- Show or not the RSSI value of the tag read;
- Show or not the read count of tag read;
- Audible reading of tag;
- Check internal temperature;

8.2.1. How to configure the reader?

The reader in the Autonomous mode must be configured via command line using ASCII commands sent through the serial port created in the Operating System when the reader is turned on.

> **NOTE:** To find out which serial port the reader will respond to, please consult 7.1. Turning on the reader

8.2.1.1. Logic of operation

The configuration of the reader via ASCII commands works with the Message / Response structure:
- A. An external host sends a command to the reader;
- B. The reader interprets and executes the command inserted in the received message;
- C. The reader responds to the external Host.

Messages and responses use ASCII characters with a carriage return terminator <CR>, which is [ENTER].
Messages can be in the upper or lower case, or both.

| NOTE: | The reader echoes each character typed in the serial terminal and each new line the reader sends the character ">" to indicate the cursor. |

8.2.1.2. Syntax of messages and responses

The following is the syntax of Host-to-Reader messages:

COMMAND VALUE<CR>	Changes the value of the command / parameter.
COMMAND<CR>	Reads the value of the command / parameter.

⚠️ **WARNING:** **Only one value must be passed to each command, the reader will not accept more than one value per command.**

The following is the syntax of Reader-to-Host response messages:

RESULT<CR>	Returns the result of the execution of the last command sent to the reader.

8.2.1.3. How to send the commands to the reader? (Windows)

The reader settings in Autonomous mode can be performed through a terminal emulation software. The best known open source and free is **Putty** whose link to Download follows below.

https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html

To configure Putty follow these steps:

1. Open Putty and select "Connection type:" "Serial", "Serial line" the port that reader generated and "Speed" 115200.
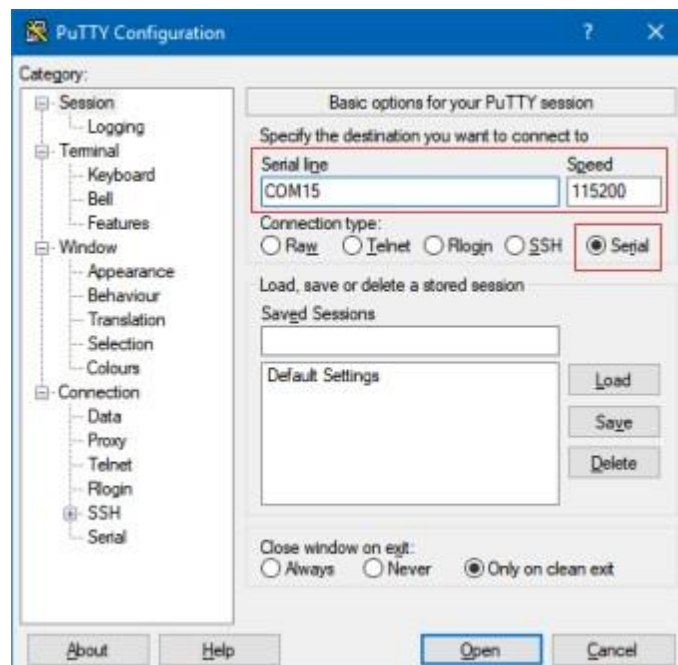
2. Click "Open" to open the terminal screen for entering commands.

8.2.1.4. How to send the commands to the reader? (Linux and Mac OS)

On Linux operating systems you can also use Putty as explained above for Windows.

For Mac OS and Linux you can use the program "screen" in a Terminal session as below.

NOTE:	To find out which serial port the reader will respond to, please consult 7.1. Turning on the reader

(Mac OS)

$ **screen /dev/tty.usbmodemXXXXXX 115200**

Where "XXXXXX" is a unique identifier number generated by the operating system.
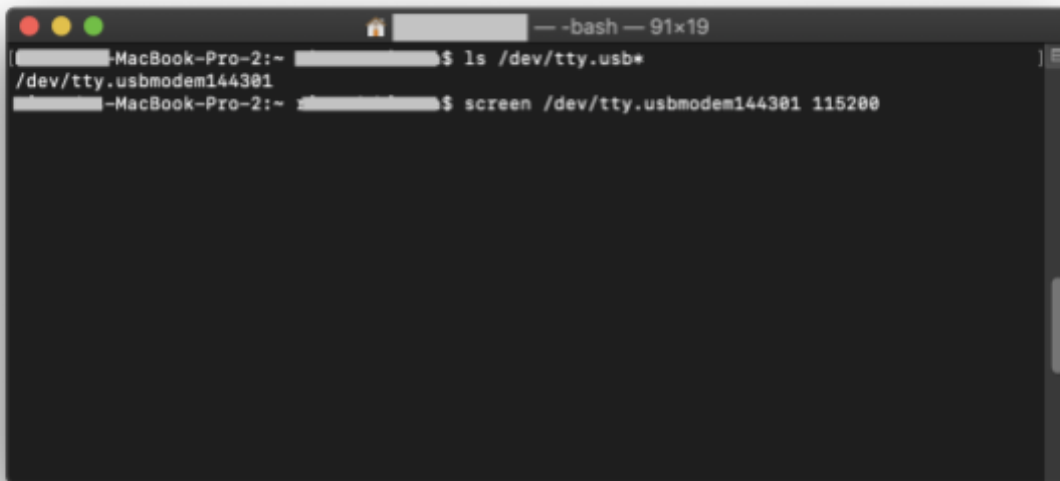
(Linux) using screen.

NOTE:	In linux to have permission to use the serial port, run the command with super user permission (sudo) or add your user to the dialout group using the following command: $ **sudo usermod -a -G dialout $USER** In order to take effect, you must restart the system.

To connect using screen:

$ **screen /dev/ttyACMx**
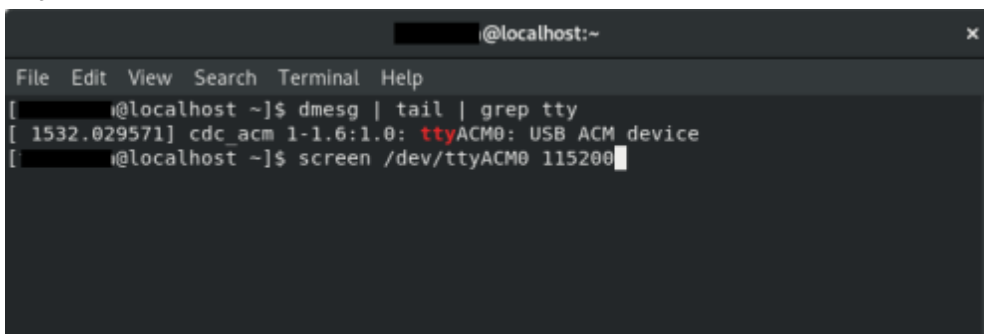
8.3. Configuration commands

8.3.1. Important tip

It is not always easy to remember the name of the command or the values accepted, so there is the **help** command that if executed without any value lists all the commands supported by the reader and to get details of a specific command just type **help command<CR>** this is a quick way to access the reader configuration information. Below is an illustration of the help command.

8.3.2. Summary of configuration commands and default values

Command	Description	Default value	Unit
readmode	Reading mode and communication interface.	hid	
readpower	Read power.	15	dBm
rfon	Read time.	250	ms
rfoff	Idle time.	250	ms
gen2blf	EPC Gen2 Backscatter Link Frequency in kHz.	250	
gen2tari	EPC Gen2 Tari, bit pulse duration.	25	us
gen2tagencoding	EPC Gen2 encoding communication tag-leitor.	m4	
gen2session	EPC Gen2 tag session.	s1	
gen2target	EPC Gen2 tag target.	a	
gen2q	EPC Gen2 amount of tag in the reading field.	dynamic	
gen2accesspassword	EPC Gen2 tag access password.	00000000	
temperature	Reader temperature in.		°C
version	Firmware version of the reader's controller.		
rfidversion	Firmware version of reader's rf module.		
rfidmodule	RF module's name.		
serialnumber	Reader serial number.		
tagop	Tag reading operation.	none	
epcdecode	EPC tag id decoding.	none	
autoidcs	Acura Autoid tags security code.	9000001	
separator	Result string separator character.	' ' (espaço)	
endofline	End of line type of result string.	windows	
keystrokespeed	Typing speed of hid interface.	fast	
rssifilter	Filter by RSSI value of the tag.	0	
reportrssi	Report the RSSI value of the read tag.	off	
reportreadcount	Reports the tag read count in the "rfon" period.	off	
enablebeep	Audible reading of tag.	on	
enablecapslock	Shortcut via "CapsLock" key to switch between Autonomous and Transparent modes.	on	
help	Help text for each command.		
summary	Summary of reader settings.		
facreset	Returns all settings to factory default.		

8.3.3. Command reply messages

Upon receiving a command, the reader executes and responds with one of the following messages:

Type	Description
Success	When the value entered is valid and different from the current recorded on the reader: **ok - parameter has been changed.<CR>** When the value entered is valid and the same as the value recorded in the reader: **ok - parameter already has this value.<CR>** When the **facreset** command is executed: **!!! Factory Reset in action !!! The reader will be restarted.<CR>**
Errors	When a non-existent command is sent to the reader: **error - command not found<CR>** When the value entered is not among those specified and accepted by the reader: **error - value out of range<CR>** When more than one value is passed to the command: **error - too many args.<CR>** When a value not supported by the RF module is passed: **error - value not supported for this RF Module.<CR>** When a value not supported by the BLF already configured is passed: **error - gen2 wrong parameters combination.<CR>** When the number of characters before <CR> is greater than 50: **error - internal buffer full.<CR>**

8.3.4. Details of configuration commands

8.3.4.1. readmode - Reader mode of operation

Command	**readmode** *value*
Function	Check (no value) or change the operating mode of the reader.
Description	The reader can function Autonomously with automatic readings, or Transparent where a Software can have complete control of the reader for tag read and write operations.
Default	**hid**
Values	● **hid** ○ (Autonomous) Keyboard emulation mode, when a tag is read the result is sent to the computer as if typed quickly. ● **hidserial** ○ (Autonomous) Keyboard emulation mode and USB Serial, the reading result is sent via keyboard emulation and also via serial communication on the USB virtual port. ● **serial** ○ (Autonomous) Serial communication mode, the reading result is sent via serial communication on the USB virtual port. ● **transparent** ○ The reader must be controlled by a Software that has full control of the reader to perform tag read/write operations.
Example	To check the operating mode: **readmode<CR>** **hid<CR>** To change the operating mode (Autonomous reporting via serial only): **readmode serial<CR>** **ok - parameter has been changed.<CR>**
Errors	When the value entered is not among those specified: **error - value out of range<CR>** When more than one value is passed to the command: **error - too many args.<CR>**

8.3.4.2. readpower - Read power

Command	**readpower** *value*
Function	Check (no value) or change the read power in dBm.
Description	The reader in Autonomous mode can have a reading power range of 0 to 15 dBm.
Default	**10 dBm**
Values	**0 .. 15**
Example	To check the read power: **readpower<CR>** **12<CR>** To change the read power to 5 dBm: **readpower 5<CR>** **ok - parameter has been changed.<CR>**
Errors	When the value entered is not among those specified: **error - value out of range<CR>** When more than one value is passed to the command: **error - too many args.<CR>**

NOTE: The reading power directly influences the reading distance of the tag, the temperature at which the reader will reach and its energy consumption, we recommend always to adjust the best reading power according to your application.

8.3.4.3. rfon - Read time

Command	**rfon** *value*
Function	Check (no value) or change the reading time in ms (milliseconds).
Description	Read time is the time the reader transmits RF energy to the antenna in an attempt to read tag. When you have many tags to read at the same time, the reading time should be increased.
Default	**250 ms**
Values	**50 .. 5000**
Example	To check the reading time: **rfon<CR>** **250<CR>** To change reading time to 500 ms (0,5 s): **rfon 500<CR>** **ok - parameter has been changed.<CR>**
Errors	When the value entered is not among those specified: **error - value out of range<CR>** When more than one value is passed to the command: **error - too many args.<CR>**

NOTE: The reading results are reported by the reader after the sum of rfon and rfoff times.

NOTE: If the application requires reading multiple tags at the same time, leave the rfon above 250ms to allow time to power all the tags in the reader field.

8.3.4.4. rfoff - Idle time

Command	**rfoff** *value*
Function	Check (no value) or change the time interval between reads in ms (milliseconds).
Description	Time duration that the reader turns off the RF energy in its antenna.
Default	**250 ms**
Values	**0 .. 1000**
Example	To check the idle time: **rfoff\<CR>** **250\<CR>** To change the idle time to 500 ms (0,5 s): **rfoff 500\<CR>** **ok - parameter has been changed.\<CR>**
Errors	When the value entered is not among those specified: **error - value out of range\<CR>** When more than one value is passed to the command: **error - too many args.\<CR>**

> **NOTE:** The reading results are reported by the reader after the sum of rfon and rfoff times.

8.3.4.5. gen2blf - EPC Gen2 Backscatter Link Frequency

Command	**gen2blf** *value*
Function	Check (no value) or change the "Backscatter Link Frequency" EPC Gen2 parameter.
Description	BLF is the data rate in KHz of the air transmission between a tag and the reader.
Default	**250 Khz**
Values	**250** **320** **640**
Notes	Changing the BLF to 640 or 320 KHz, the value of the Tari and tag encoding will automatically be changed to 6.25 us and fm0 respectively.
Example	To check the BLF: **gen2blf<CR>** **250<CR>** To change BLF to 640 KHz: **gen2blf 640<CR>** **ok - parameter has been changed.<CR>**
Errors	When the value entered is not among those specified: **error - value out of range<CR>** When more than one value is passed to the command: **error - too many args.<CR>**

⚠ **WARNING:** **Changing gen2blf will automatically change other EPC gen2 parameters, which are gen2tari and gen2tagencoding.**

8.3.4.6. gen2tari - EPC Gen2 Tari bit pulse duration

Command	**gen2tari** *value*
Function	Check (no value) or change the EPC Gen2 "Tari"parameter.
Description	Tari is the time duration in us (microseconds) of the pulse of each bit transmitted by the tag.
Default	**25 us**
Values	**6.25** **12.5** **25**
Notes	The 12.5 us and 25 us values can only be set if gen2blf is set to 250 KHz. The 6.25 us and 12.5 us values are supported only by the built-in micro-module RF module, which can be verified by the **rfidmodule** command.
Example	To check Tari value: **gen2tari<CR>** **25<CR>** To change Tari to 12.5 us: **gen2tari 12.5<CR>** **ok - parameter has been changed.<CR>**
Errors	When the value entered is not among those specified: **error - value out of range<CR>** When more than one value is passed to the command: **error - too many args.<CR>** When a value not supported by the RF module is entered: **error - value not supported for this RF Module.<CR>** When a value not supported by the current BLF configured is entered: **error - gen2 wrong parameters combination.<CR>**

⚠ **WARNING: The Tari value should be combined with the BLF value.**

8.3.4.7. gen2tagencoding - EPC Gen2 tag encoding

Command	**gen2tagencoding** *value*
Function	Check (no value) or change the "Tag data encoding" of EPC Gen2 protocol.
Description	It is the type of data encoding sent through the air by the tag.
Default	**m4 (Miller 4)**
Values	**fm0** **m2** **m4** **m8**
Notes	m2, m4 and m8 can only be configured if the blf is 250 KHz. fm0 has a low reading sensitivity rate but a higher speed in decoding each bit. m8 has a high reading sensitivity rate but a lower decoding speed of each bit.
Example	To check data encoding: **gen2tagencoding<CR>** **m4<CR>** To change the tag encoding to m8: **gen2tagencoding m8<CR>** **ok - parameter has been changed.<CR>**
Errors	When the value entered is not among those specified: **error - value out of range<CR>** When more than one value is passed to the command: **error - too many args.<CR>** When a value not supported by the current BLF configured is entered: **error - gen2 wrong parameters combination.<CR>**

> ⚠ **WARNING: The Tari value should be combined with the BLF value.**

8.3.4.8. gen2session - EPC Gen2 tag session

Command	**gen2session** *value*
Function	Check (no value) or change the tag "Session".
Description	Configure in which session the reader will read a tag.
Default	**s1**
Values	**s0** **s1** **s2** **s3**
Notes	The combination of values of gen2session and gen2target reflects on the read rate of a same tag.
Example	To check the current Session: **gen2session<CR>** **s1<CR>** To change session to s0: **gen2session s0<CR>** **ok - parameter has been changed.<CR>**
Errors	When the value entered is not among those specified: **error - value out of range<CR>** When more than one value is passed to the command: **error - too many args.<CR>**

> **NOTE:** Use gen2session and gen2target to manipulate the read rate of the same tag as described in 8.5. Read rate of same tag.

8.3.4.9. gen2target - EPC Gen2 tag target

Command	**gen2target** *value*
Function	Check (no value) or change the tag "Target".
Description	It is the flag of each session of the tag.
Default	**a**

Values	**a**
	
	    Reading tags with session flag in state A.
	
	b
	
	    Reading tags with session flag in state B.
	
	ab
	
	    Read tags with session flag in state A, then in B.
	
	ba
	
	    Read tags with session flag in state B, then in A.

Notes	The combination of values of gen2session and gen2target reflects on the read rate of a same tag.
Example	To check current Target: **gen2target<CR>** **a<CR>** To change target to ab: **gen2target ab<CR>** **ok - parameter has been changed.<CR>**
Errors	When the value entered is not among those specified: **error - value out of range<CR>** When more than one value is passed to the command: **error - too many args.<CR>**

NOTE: Use gen2session and gen2target to manipulate the read rate of the same tag as described in 8.5. Read rate of same tag.

8.3.4.10. gen2q - EPC Gen2 number of tags in the reading field

Command	**gen2q** *value*
Function	Check (no value) or change the "Q" value.
Description	Q it is a numeric value that the reader uses to regulate the probability of response of a tag, used for anti-collision when several tags are in the reader's reading field. 2^Q equals the number of tags to be read at the same time.
Default	**dynamic**

Values	**dynamic**
	      The reader automatically adjusts the Q value during readings.
	
	0 .. 15
	      Fixed value for 2^Q number of tags in reader field.

Notes	If the number of tags to be read at the same time varies, leave the value q in dynamic, so the reader automatically adjusts the best value for each reading.
Example	To check current Q value: **gen2q<CR>** **dynamic<CR>** To change Q value to 3, with 8 tags in reading field: **gen2q 3<CR>** **ok - parameter has been changed.<CR>**
Errors	When the value entered is not among those specified: **error - value out of range<CR>** When more than one value is passed to the command: **error - too many args.<CR>**

NOTE: Use gen2q as dynamic if the number of tags in the read field varies or is not known.

8.3.4.11. gen2accesspassword - EPC Gen2 tag access password

Command	**gen2accesspassword** *value*
Function	Check (no value) or change the tag "Access password".
Description	It is the 32-bit (8-digit) password of the tag, in hexadecimal.
Default	**00000000**
Values	**00000000 .. ffffffff**
Notes	To read the reserved memory bank of the tag with the **tagop** command, the password must be the same as the tag.
Example	To check current Access password: **gen2accesspassword<CR>** **00000000<CR>** To change password to 12345678: **gen2accesspassword 12345678<CR>** **ok - parameter has been changed.<CR>**
Errors	When the value entered is not among those specified: **error - value out of range<CR>** When more than one value is passed to the command: **error - too many args.<CR>**

NOTA: To read the protected memory region of the tag, Reserved, the tag access password must be informed to the reader through this command.

8.3.4.12. tagop - Tag reading operation

Command	**tagop** *value*
Function	Check (no value) or change the tag reading operation.
Description	With the reading operation it is possible to read any memory bank of the tag and also tags of the Acura Autoid System.
Default	**none**

Values	**none**
	        No memory bank read operation.
	autoid
	        Read the Acura Autoid System tags.
	reserved:wp:wl
	        Read tag's Reserved memory bank.
	epc:wp:wl
	        Read tag's EPC memory bank.
	tid:wp:wl
	        Read tag's Tid memory bank.
	user:wp:wl
	        Read tag's User memory bank.

Notes	**wp** - word pointer,indicates the word position of the read operation in the specified memory bank. **wl** - word length, indicates the number of words to be read in the specified memory bank. Maximum value for wp and wl is 32. To read the entire memory bank, simply specify zero for wp and wl.
Example	To read the entire contents of the User memory bank of the tag: **tagop user:0:0<CR>** **ok - parameter has been changed.<CR>** To read four words from the Tid memory bank starting after the second word: **tagop tid:2:4<CR>** **ok - parameter has been changed.<CR>** No specific tag memory bank is required, only the EPC id result: **tagop none<CR>** **ok - parameter has been changed.<CR>**
Errors	When the value entered is not among those specified: **error - value out of range<CR>** When more than one value is passed to the command: **error - too many args.<CR>**

> **NOTE:**   To read the entire memory bank, simply specify **zero for wp and wl**.

8.3.4.13. epcdecode - Tag's EPC id decoding

Command	**epcdecode** *value*
Function	Check (no value) or change the decoding type of tag's EPC id.
Description	The tag to be read may be coded for a certain application, e.g., for checkout applications the tag will be encoded in the GS1 standard, for maintenance applications the tag may be ASCII encoded. Tag's EPC id decoding returns a readable result for each application.
Default	**none**

Values	**none**
	    No decoding is performed and the EPC id result of the tag is reported in hexadecimal.
	gs1string
	    GS1 standard "EPC TDS 1.11" and returns the "GS1 element string".
	gs1epcuri
	    GS1 standard "EPC TDS 1.11" and returns the "EPC Tag URI".
	gs1epcpureuri
	    GS1 standard "EPC TDS 1.11" and returns the "EPC Pure Identity URI".
	gs1gtin13
	    GS1 standard "EPC TDS 1.11" and returns the GTIN13 code without item serial number.
	gs1sgtin13
	    GS1 standard "EPC TDS 1.11" and returns the GTIN13 code + item serial number..
	ascii
	    Use the Ascii table to decode the tag's EPC id..
	wiegand26
	    Decodes the last 3 bytes of the EPC id in format "Sitecode + ID".
	magstripe
	    Decodes the last 4 bytes of the EPC id in decimal format.

Notes	For the GS1 standard, the sgtin-96 and sgtin-198 schemas are supported. The wiegand26 and magstripe values can be used in conjunction with the autoid type **tagop**.
Example	To read the GTIN13 + serial number of a tag encoded in the GS1 standard: **epcdecode gs1sgtin13\<CR\>** **ok - parameter has been changed.\<CR\>** To read the GS1 element string from a tag encoded in the GS1 standard: **epcdecode gs1string\<CR\>** **ok - parameter has been changed.\<CR\>**
Errors	When the value entered is not among those specified: **error - value out of range\<CR\>** When more than one value is passed to the command: **error - too many args.\<CR\>**

8.3.4.14. autoidcs - Acura Autoid System Security Code

Command	**autoidcs** *value*
Function	Check (no value) or change the Acura Autoid System Security Code.
Description	It's a six-digit number that authenticate the tags of Acura Autoid System.
Default	**900001**
Values	**000001 .. 999999**
Notes	This command should only be used if the **tagop** is autoid.
Example	To check autoidcs: **autoidcs<CR>** **900001<CR>** To change the security code to 900158: **autoidcs 900158<CR>** **ok - parameter has been changed.<CR>**
Errors	When the value entered is not among those specified: **error - value out of range<CR>** When more than one value is passed to the command: **error - too many args.<CR>**

NOTA: This command is only used if the application is to read the tags of the Acura Autoid System, **tagop** configured as autoid.

NOTA: The value of the security code is printed on the tag, please consult XXXXXXXX.

8.3.4.15. separator - Separation character of result string

Command	**separator** *value*
Function	Check (no value) or change the separating character of the fields in the reading result string.
Description	It is a character of the Ascii table, which separates the fields in the tag reading result string.
Default	**' ' (espace char)**

Values	**space**
	       To specify space char as separator.
	'char'
	       Any readable character in the Ascii table.

Notes	For space, you should write "space" as the value for the command.
Example	To check the current separator: **separator<CR>** **space<CR>** To change separator to '|' char: **separator |<CR>** **ok - parameter has been changed.<CR>**
Errors	When the value entered is not among those specified: **error - value out of range<CR>** When more than one value is passed to the command: **error - too many args.<CR>**

NOTE: The separator character in conjunction with the end of line must be used to divide the result string by the Software that will receive that string.

8.3.4.16. endofline - Result string end of line

Command	**endofline** *value*
Function	Check (no value) or change the end of line type.
Description	It is one or more unreadable characters that are at the end of the result string and represent the end of the line for the Operating System.
Default	**windows**
Values	**windows:qt** 　　　　Windows default end of line <CR><LF>. **unix:qt** 　　　　Linux default end of line <LF>. **macintosh:qt** 　　　　Mac default end of line<CR>. **tab:qt** 　　　　Tabulation char as end of line. **none** 　　　　No end of line in result string.
Notes	**qt** - End of line repeat count, values 1 to 10. If **:qt** is omitted, only one end of line will be in the result string.
Example	To check the current end of line type: **endofline<CR>** **windows<CR>** To change the end of line to be Tabulation and repeat 3 times: **endofline tab:3<CR>** **ok - parameter has been changed.<CR>** To change the end of line to Windows standard and repeat 5 times: **endofline windows:5<CR>** **ok - parameter has been changed.<CR>**
Errors	When the value entered is not among those specified: **error - value out of range<CR>** When more than one value is passed to the command: **error - too many args.<CR>**

NOTE: The separator character in conjunction with the end of line must be used to divide the result string by the Software that will receive that string.

8.3.4.17. keystrokespeed - Keyboard emulation typing speed

Command	**keystrokespeed** *value*
Function	Check (no value) or change the keyboard emulation typing speed.
Description	The typing speed in keyboard emulation, when the readmode is as hid or hidserial, can be changed to three different values.
Default	**fast**

Values	**fast**
	    Fast typing emulation, period between typing T = 4ms.
	medium
	    Medium typing emulation, period between typing T = 32ms.
	slow
	    Slow typing emulation, period between typing T = 32ms.

Example	To check the current typing emulation speed: **keystrokespeed<CR>** **fast<CR>** To change the speed to slow : **keystrokespeed slow<CR>** **ok - parameter has been changed.<CR>**
Errors	When the value entered is not among those specified: **error - value out of range<CR>** When more than one value is passed to the command: **error - too many args.<CR>**

NOTE: This value should be changed only if the host that will receive the result string via keyboard emulation presents problems due to the typing speed exerted by the reader.

8.3.4.18. rssifilter - RSSI read filter

Command	**rssifilter** *value*
Function	Check (no value) or change the RSSI cut value to filter readings.
Description	The Tag Signal Level (RSSI) indicates the strength of tag's received signal. You can estimate its distance from the antenna, a RSSI (Received Signal Strength Indication) value near to Zero means that the signal is stronger and the tag is closer. Negative values further from zero means the signal is weaker and the tag is further. The reader will report only tags with a RSSI equal or higher than the value in this field. The RSSI is measured in dBm and assumes negatives values, which means values lesser than 1mW.
Default	**0**
Values	**0..99**
Notes	To disable the filter, just specify 0. Do not type the negative signal, only the absolute values.
Example	To check the current filter value: **rssifilter<CR>** **0<CR>** To change the filter to -45dBm : **rssifilter 45<CR>** **ok - parameter has been changed.<CR>**
Errors	When the value entered is not among those specified: **error - value out of range<CR>** When more than one value is passed to the command: **error - too many args.<CR>**

NOTE:	The RSSI tag filter is used to restrict unwanted areas of reading around the reader. Must always be used in conjunction with read power.

8.3.4.19. reportrssi - Report the RSSI value of the read tag

Command	**reportrssi** *value*
Function	Check (no value) or change the RSSI value sending of the read tag.
Description	Enables or disables sending the RSSI value of the read tag in the result string.
Default	**off**
Values	**on** **off**
Notes	The value of RSSI is reported with the negative sign '-'.
Example	To check the current state of the flag: **reportrssi<CR>** **off<CR>** To change the value to have the RSSI value in the result string: **reportrssi on<CR>** **ok - parameter has been changed.<CR>**
Errors	When the value entered is not among those specified: **error - value out of range<CR>** When more than one value is passed to the command: **error - too many args.<CR>**

8.3.4.20. reportreadcount - Reports the read count of a tag

Command	**reportreadcount** *value*
Function	Check (no value) or change the sending of read count of a tag..
Description	Enables or disables sending the amount of tag reading in the "rfon" period in the result string.
Default	**off**
Values	**on** **off**
Notes	The maximum value of the readcount reported is 255.
Example	To check the current flag value: **reportreadcount\<CR\>** **off\<CR\>** To change the flag to send the readcount information in the result string: **reportreadcount on\<CR\>** **ok - parameter has been changed.\<CR\>**
Errors	When the value entered is not among those specified: **error - value out of range\<CR\>** When more than one value is passed to the command: **error - too many args.\<CR\>**

8.3.4.21. enablebeep - Beep feedback

Command	**enablebeep** *value*
Function	Check (no value) or change the beep feedback at each tag reading.
Description	Enables or disables audible warning at each tag reading.
Default	**on**
Values	**on** **off**
Example	To check the current value: **enablebeep<CR>** **off<CR>** To disable the beep feedback: **enablebeep off<CR>** **ok - parameter has been changed.<CR>**
Errors	When the value entered is not among those specified: **error - value out of range<CR>** When more than one value is passed to the command: **error - too many args.<CR>**

8.3.4.22. enablecapslock - Shortcut for changing the reading mode

Command	**enablecapslock** *value*
Function	Check (no value) or change the use of the shortcut to change the reading mode.
Description	Enables or disables the use of the "CapsLock" button 5x quick typing, to switch from Autonomous mode to Transparent mode and vice versa.
Default	**off**
Values	**on** **off**
Notes	This feature works only on Windows and Linux operating systems. Alternatively to exit Transparent mode is to send the command "**$ off <CR>**" to the reader.
Example	To check the current value: **enablecapslock<CR>** **off<CR>** To enable the shortcut mode switch: **enablecapslock on<CR>** **ok - parameter has been changed.<CR>**
Errors	When the value entered is not among those specified: **error - value out of range<CR>** When more than one value is passed to the command: **error - too many args.<CR>**

> **NOTE:** For details on changing the reader operating mode, please refer to 8.4. Switching from Autonomous to Transparent mode and vice versa.

8.3.4.23. help - Help on commands

Command	**help** *value*
Function	Lists all commands (no value) or shows detailed help for the entered command.
Description	For detailed help on a given command simply enter: "help command<CR>".
Example	To check the commands supported by the reader: **help<CR>** **lista de comando...<CR>** For detailed help on the **tagop** command: **help tagop<CR>** **Command: tagop arg<CR>** **Description: Get (no arguments) or set the read tag operation.<CR>** **Arguments: reserved:wp:wl Reserved bank.*<CR>** ** epc:wp:wl EPC bank.*<CR>** ** tid:wp:wl TID bank.*<CR>** ** user:wp:wl User bank.*<CR>** ** autoid Autoid tags.<CR>** ** none No tagop.<CR>** ** * wp - word pointer, wl - word length, max 32, menbank:0:0 entire bank.** **Default: none<CR>**
Errors	When the value entered is not among those specified: **error - value out of range<CR>** When more than one value is passed to the command: **error - too many args.<CR>**

8.3.4.24. summary - Summary of the values set in the reader

Command	**summary**
Function	List all values set in the reader
Errors	When more than one value is passed to the command: **error - too many args.<CR>**

8.3.4.25. facreset - Factory reset

Command	**facreset**
Function	Returns all values to the factory default.
Errors	When more than one value is passed to the command: **error - too many args.<CR>**

	Beep and green and red LED blinks fast simultaneously for 5s to warn that it is in the process of factory reset.

8.4. Switching from Autonomous to Transparent mode and vice versa

To change the operating mode of the reader the following options can be used:

A. Using the **readmode** command, examples:

   a. The reader is in Autonomous mode and the "**readmode transparent<CR>**" command is sent, so the reader enters Transparent mode of operation.

   b. The reader is in Transparent mode and the "**$ off <CR>**" command is sent, so the reader enters the Autonomous mode with hid interface.

NOTE:	When in Transparent mode, the reader does not echo the characters entered on the serial terminal.

B. Use the "CapsLock" key (Windows and Linux only), for example:

   a. Close any serial connection with the reader;

   b. To switch between one operating mode and another, simply click the Caps Lock button 5 times fast in sequence and check the leds of the reader to see the change.

⚠️ **WARNING:**	**The Caps Lock shortcut for switching the operating mode only works if and only if the reader's serial port is not connected (not being used by some Software).**

8.5. Read rate of same tag

The reading rate of the same tag must be handled through the following commands:
- **gen2session**
- **gen2target**
- **rfon**
- **rfoff**

The commands mentioned above change the parameters of the UHF EPC Gen2 protocol and have an effect on the communication between the tag and reader. Combinations of values for the Session and Target of the tag reflect on how often a tag that has already responded to an inventory will respond again, in other words, how fast a tag reread.
Examples:

A. I would like to read only once the EPC id of a tag and that same EPC id is not repeated while the tag is in the reader's reading field. Set as follows
gen2session s2<CR>
gen2target a<CR>

B. I would like to read the same tag repeatedly while it is in the reader field. Configure as follows:
gen2session s0<CR>
gen2target ab<CR>

In this case, what will determine the repetition rate of reading are the rfon and rfoff commands, i.e., the sum of the times will dictate the repetition rate. The highest possible read rate is every 50ms:
rfon 50<CR>
rfoff 0<CR>

⚠	**WARNING:**	**The tag reading rate can be much faster than the transmission rate of the results by keyboard emulation (hid), i.e., if the chosen reading rate is high and the interface chosen is the keyboard emulation, the reader will store the reading result and will be transmitting even after the tag has been removed from the reading field.**

8.6. Usage examples

Below are some examples of using the reader in some applications with the details of the configuration commands.

8.6.1. Checkout - GS1/EAN standard tags

For applications using UHF EPC Gen2 tags encoded according to the GS1 EPC Tag Data Standards (TDS) 1.11 standard, the AC01 V2 reader can be used and supports the following tag encodings:

- **sgtin-96**
- **sgtin-198**

The results of reading the encoded tag can be sent in the following formats:

- **GS1 element string**, e.g., "(01)03608449920322(21)1540341";
- **EPC Tag URI**, e.g., "urn:epc:tag:sgtin-96:1.360844.0992032.1540341";
- **EPC Pure Identity URI**, e.g., "urn:epc:id:sgtin:360844.0992032.1540341";
- **GTIN13 without serial number**, e.g., "3608449920322";
- **GTIN13 + serial number**, e.g., "3608449920322 1540341";

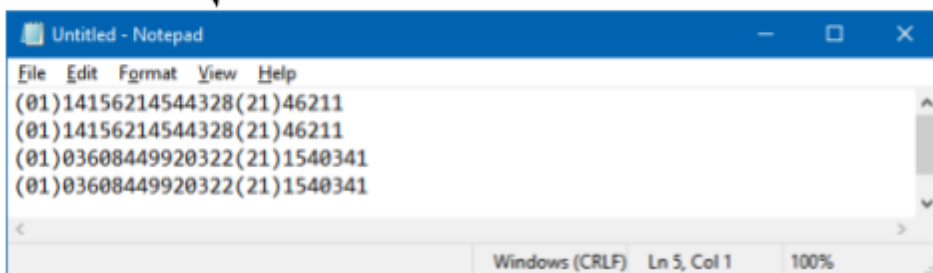### 8.6.1.1. Read tag encoded in SGTIN-96/198 and show result in GS1 format element string

Configure the reader as follows:

epcdecode gs1string<CR>

(to decode sgtin-96 or sgtin-198 and send in GS1 element string format)
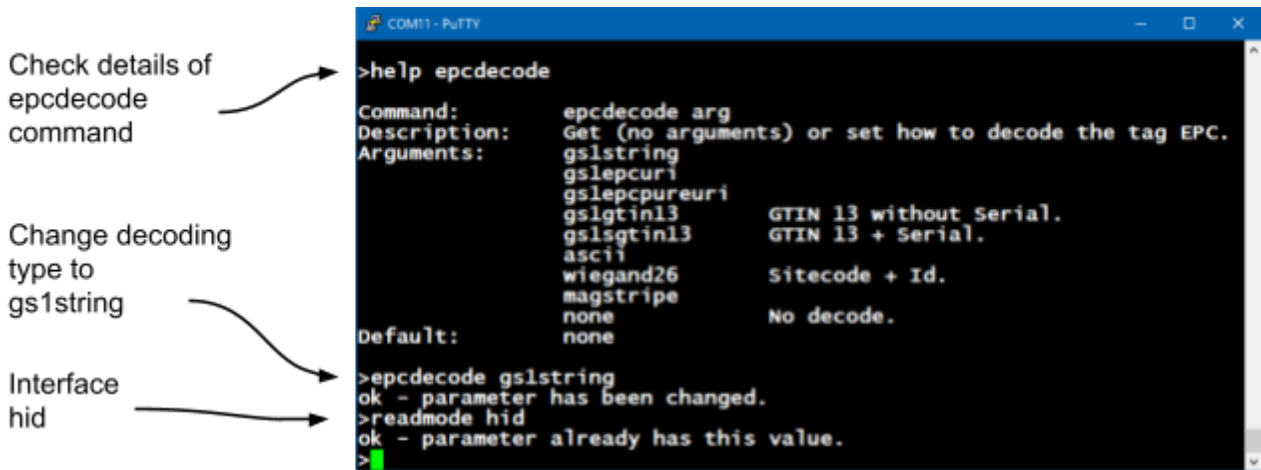
8.6.1.2. Read tag encoded in SGTIN-96/198 and show result in GS1 format epc tag uri

Configure the reader as follows:
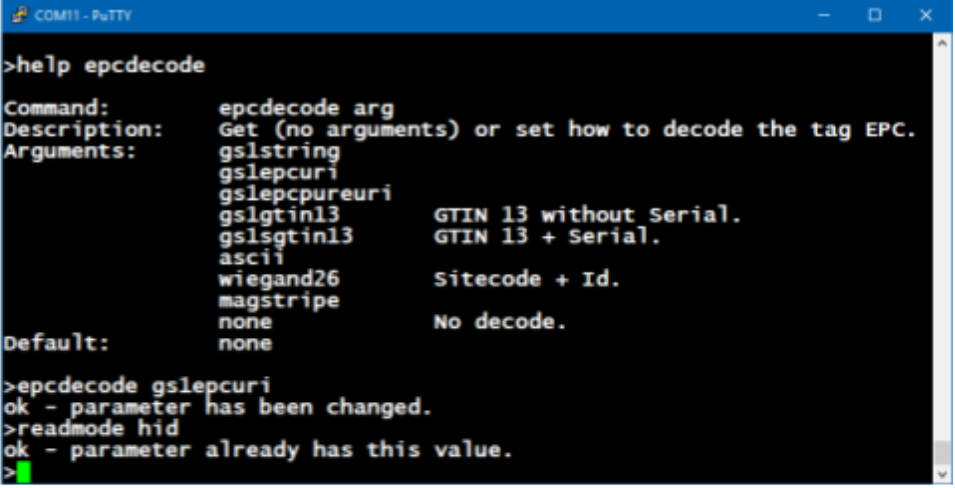
epcdecode gs1epcuri<CR>
(to decode sgtin-96 or sgtin-198 and send in GS1 EPC Tag URI format)

Check details of
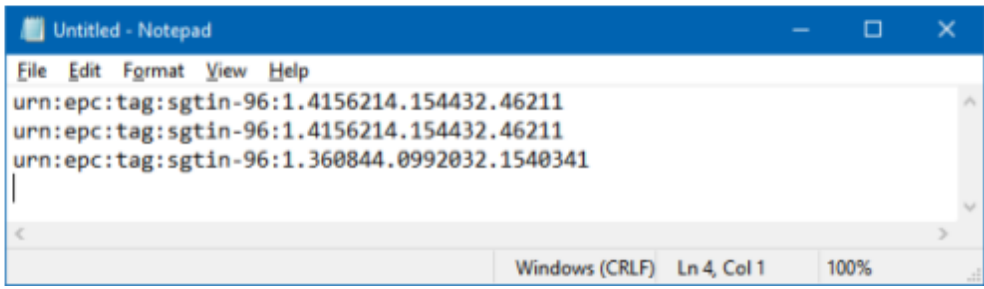epcdecode
command

Change decode to
gs1epcuri

Interface
hid

Reading results

```
>help epcdecode

Command:        epcdecode arg
Description:    Get (no arguments) or set how to decode the tag EPC.
Arguments:      gs1string
                gs1epcuri
                gs1epcpureuri
                gs1gtin13        GTIN 13 without Serial.
                gs1sgtin13       GTIN 13 + Serial.
                ascii
                wiegand26        Sitecode + Id.
                magstripe
                none             No decode.
Default:        none

>epcdecode gs1epcuri
ok - parameter has been changed.
>readmode hid
ok - parameter already has this value.
>
```

```
urn:epc:tag:sgtin-96:1.4156214.154432.46211
urn:epc:tag:sgtin-96:1.4156214.154432.46211
urn:epc:tag:sgtin-96:1.360844.0992032.1540341
```

For other formats, just use the epcdecode command with the equivalent format that can be viewed in
8.3.4.13. epcdecode - Tag's EPC id decoding.

⚠ WARNING:	The keyboard / language layout configured in the Operating System can change the ":" character of the epcdoce result string type gs1epcuri and gs1pureepcuri.

8.6.2. Read TID memory bank and EPC id tag

With the AC01 v2 reader it is possible to read any memory bank of the EPC gen2 tag (reserved, epc, tid and user), simply set up the **tagop** command on which bank to read together with the initial position and size of the data (in word) that you want to read.

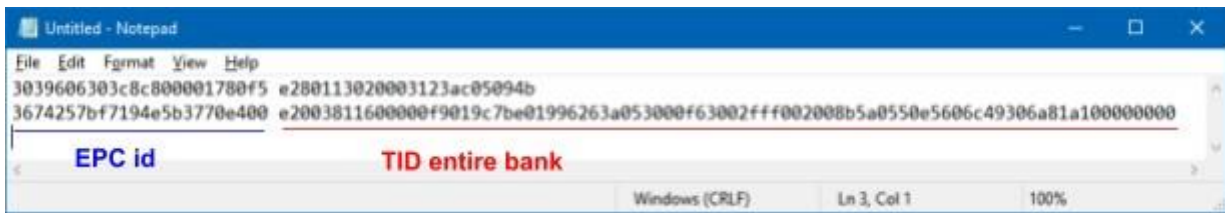The syntax of the **tagop** command is explained in detail in 8.3.4.12. tagop - Tag reading operation.

Examples:

    A. Read the entire TID memory bank plus the EPC id in hexadecimal:

        tagop tid:0:0<CR>

        epcdecode none<CR>

        Example of reading two tags with different sizes of the TID memory bank:

    B. Read from the TID bank 4 words by skipping the first 2 words and decoding the EPC id by the ASCII table:

        tagop tid:2:4<CR>

        epcdecode ascii<CR>

        Example of reading two tags:

In addition to the TID bank, the RESERVED, USER and EPC banks can also be read in conjunction with the EPC id of the tag.

NOTE:	The counting unit in the EPC Gen2 standard is word, which is equivalent to 2 bytes or 16 bits.

NOTE:	To read the Reserved memory bank of a tag with non default access password, the correct password must be set in reader using the command **gen2accesspassword**.

NOTE:	Further details of the commands in 8.3. Configuration commands.

Part 3 - Transparent mode

Reader in Transparent mode

Software development and Programming

Software demo - URA

Controlling Led e Beep

9. Reader in Transparent mode

The reader in Transparent mode responds to a software developed using the Mercury API, which is a software development package for languages C # /. Net, Java and C, with this development package it is possible to make software that can have total control of the reader for reading and writing UHF tags EPC Gen2 (ISO 18000-6C) and UHF EPC Gen2 V2 (ISO 18000-63).

Visual feedback when in Transparent mode:

Green LED flashes slowly and Red LED off
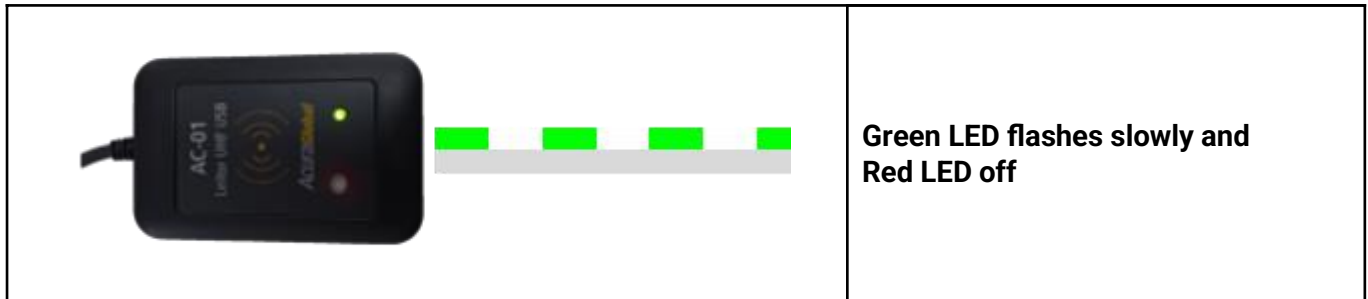
The default mode operation is Autonomous mode, to switch to Transparent mode, please refer to:
8.4. Switching from Autonomous to Transparent mode and vice versa;
8.3.4.1. readmode - Reader mode of operation.

The reader's red LED and beep is controlled by the User Software through the manipulation of two GPIOs available on MercuryAPI, more details in 9.2.2. Controlling the Red LED and Beep. Communication between the User Software and the reader is serial via the port created by the Operating System at the time the reader is connected to the USB port to identify such port, please consult 7.1. Turning on the reader.

> **NOTE:** In Transparent mode, the keyboard-type HID interface is not used by the reader.

9.1. Software Development and Programming

The last and recommended version for development can be found in:
https://www.jadaktech.com/download-product-pdf-sf?download_file=https://www.jadaktech.com/wp-content/uploads/2018/02/mercuryapi-1.31.1.36.zip&prodID=14668

In the API package there are several examples and source codes in all languages.
This development package has no additional cost and is open source.

Full API documentation:
https://www.jadaktech.com/download-product-pdf-sf?download_file=https://www.jadaktech.com/wp-content/uploads/2018/02/MercuryAPI_ProgrammerGuide_for_v1.27.3.pdf&prodID=14668

9.2. Universal Reader Assistant - URA

The reader in Transparent mode can be tested using Jadak's "Universal Reader Assistant" software, the software download and its manual are available at the links below:

User guide:
https://www.jadaktech.com/download-product-pdf-sf?download_file=https://www.jadaktech.com/wp-content/uploads/2018/02/URA2.8-UserGuide-09RevB.pdf&prodID=14685
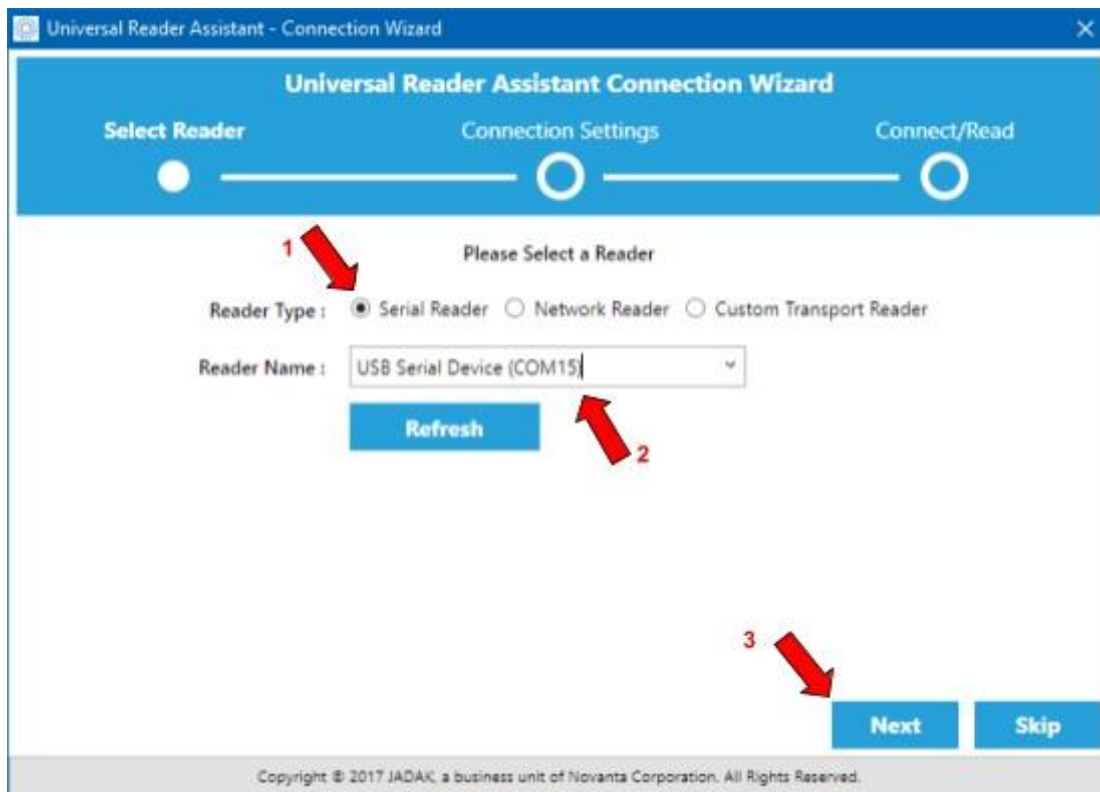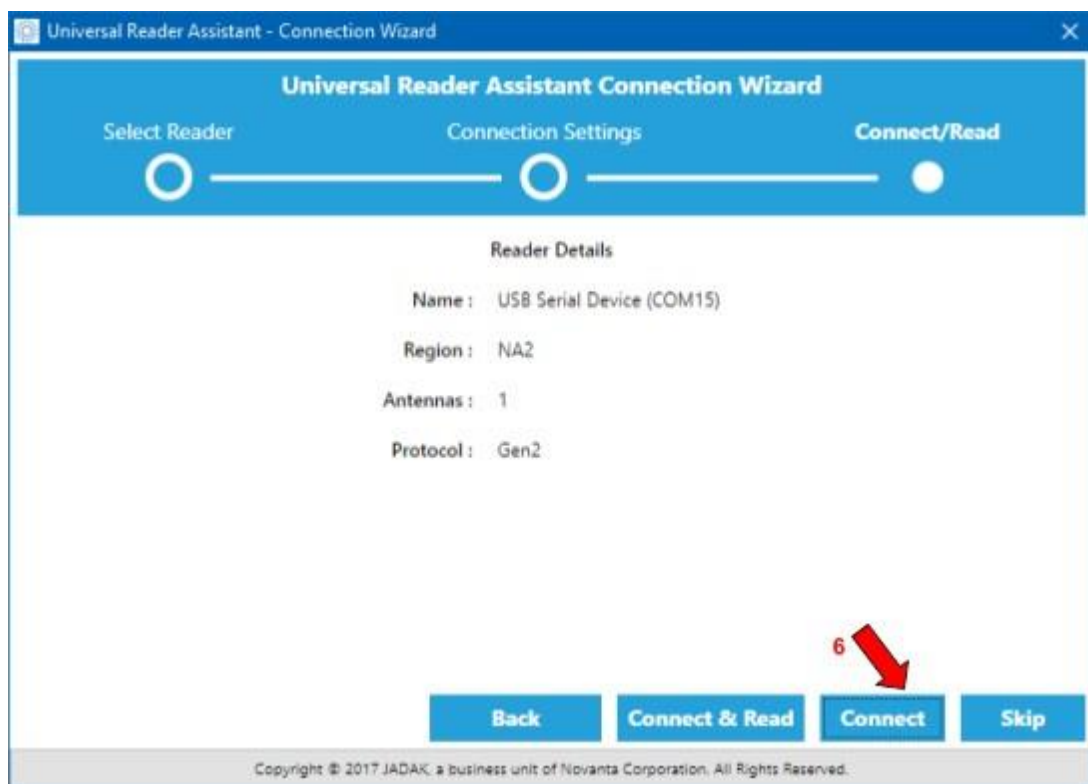
Software URA:
https://www.jadaktech.com/wp-content/uploads/2018/02/URAx64-v4.1.36.zip

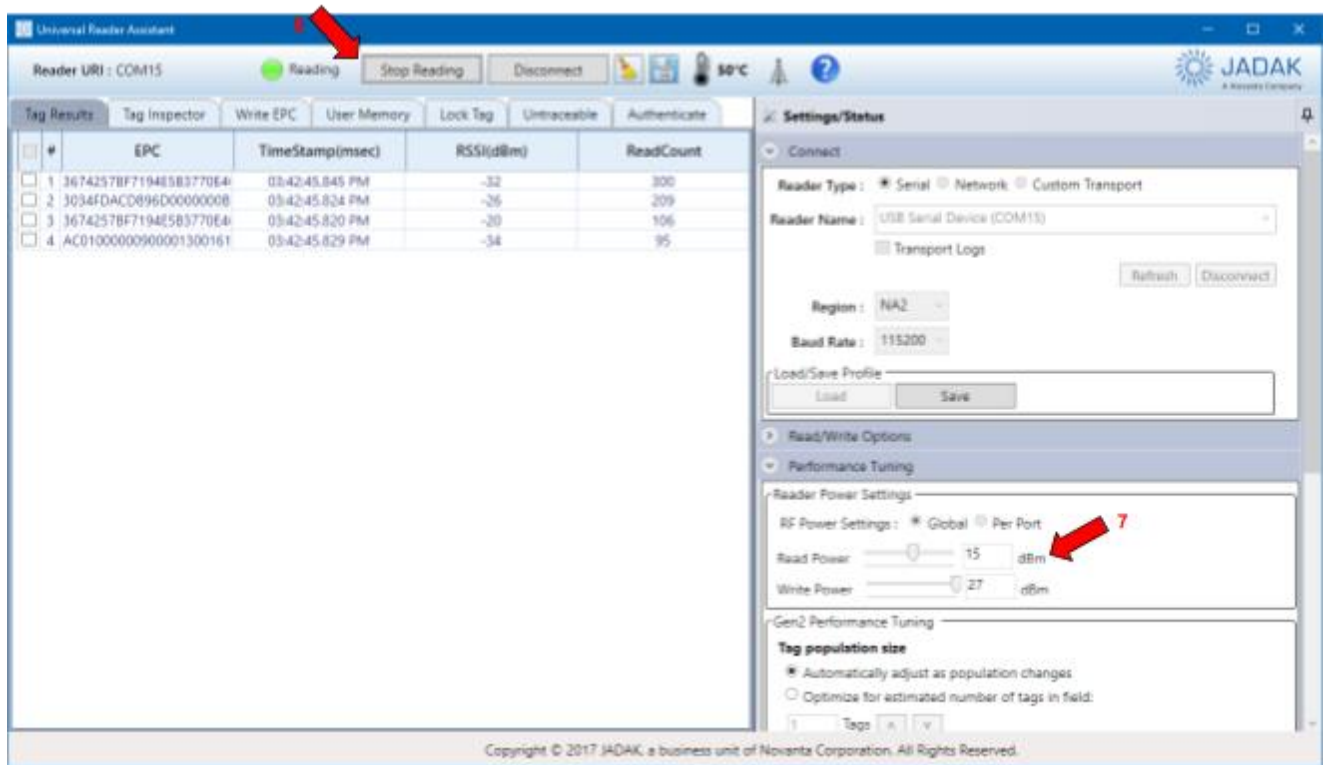### 9.2.1. Demo Software Instructions - URA

NOTE:	Make sure the reader is in Transparent mode. Please consult 9. Reader in Transparent mode.

Run the "Universal Reader Assistant".

| ⚠	**WARNING:**	**After connected, decrease the read power to 15dBm if the reader power is only from a USB port and then Read.**

9.2.2. Controlling the Red LED and Beep

In Transparent mode, the Red Led and Beep must be driven by the User Software using MercuryAPI. The following is the usage map for the respective GPIOs:

Line #	Feedback	Activation (ON)
GPIO1	Red LED	High (high level)
GPIO2	Beep	High (high level)

Below is an image using the URA to manipulate the red led and beep of the reader in Transparent mode.
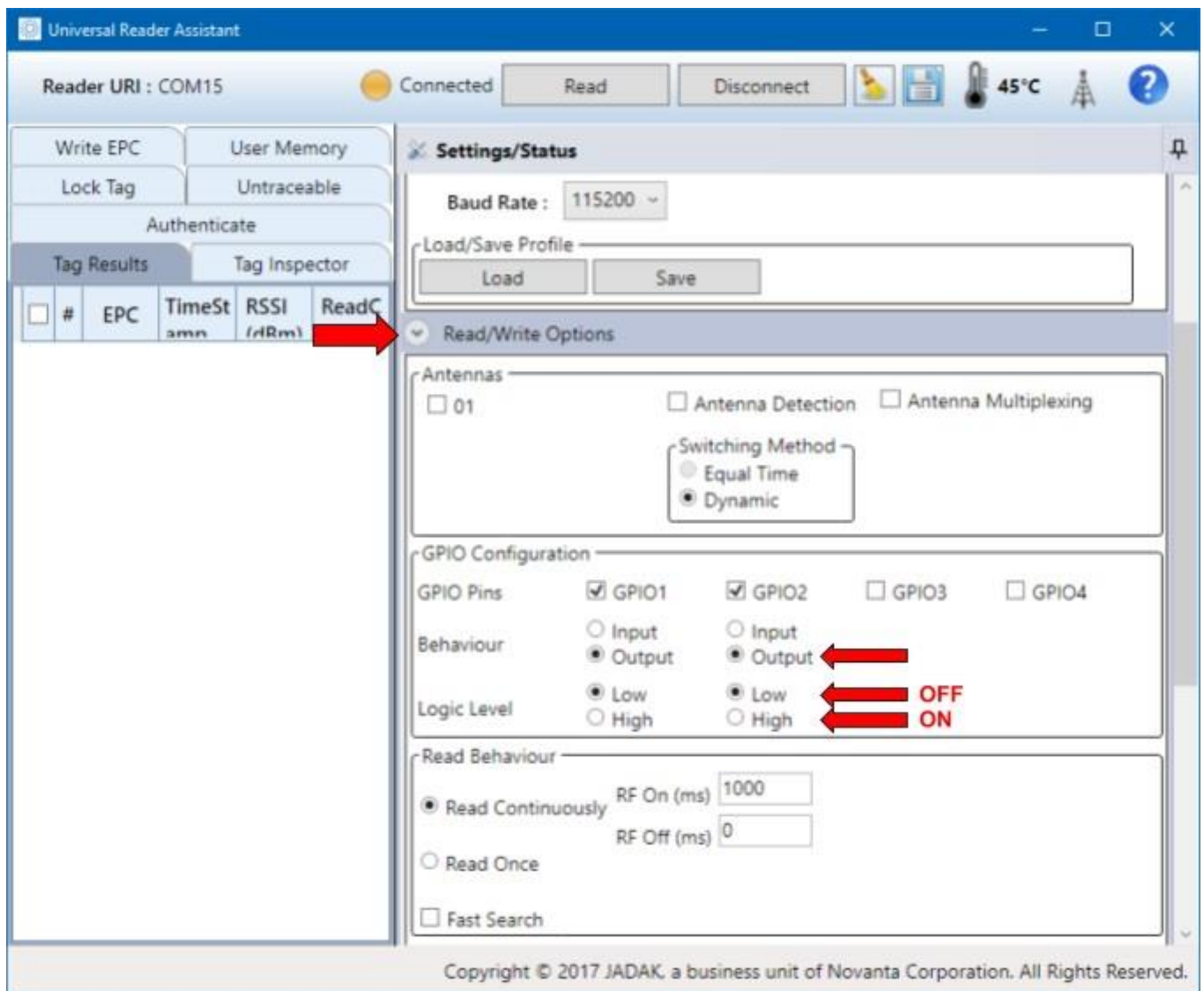
NOTE:	In the MercuryAPI file package there is an example for manipulating the GPIO of the reader called "gpiocommands", for all languages.

ACURA Global is the pioneer in the Radio Frequency Identification (RFID) market in Brazil and Latin America, and has successfully pioneered, since the late 1990s, its wide adoption scale in the most diverse sectors of the economy, from mining to steel, from agriculture to food processing, from logistics to retail, from transport to the distribution chain, from control access to asset management. Promoter of new technologies, innovative, agile and focused on viability of cutting-edge projects.

Commercial and Development Office
Av. Antártico, 381 14° andar
São Bernardo do Campo – SP
CEP: 09726-150
11 3028-4600

Factory MG
Av. Nicolau Cesarino, 4197, Extrema - MG, 37640-000

www.ACURA.com.br