ELECTRONICA, MECANICA Y CONTROL, S.A.

RealSimulator

User Guide

TUSBA TQS R2

Throttle USB Adapter

TUSBA TQS R2 - User Guide v1.07.1

Avda. Manoteras, 22 – nave 42
28050 – Madrid (Spain)

Phone: (+34) 91 383 83 25

Email: technic@realsimulator.com

Contents

PICTURE

DESCRIPTION

T.USB.A is the acronym for Throttle USB Adapter. This special version has been designed for the F16 Cougar TQS and let you connect your loved TQS to any standard USB socket.

TUSBA uses standard HID drivers included in your installed operating system, no matter if x32 or x64, XP, W7 or other higher MS operating system, TUSBA will work in all situations.

TUSBA has been designed with state of the art last generation microcomputer and has been adjusted to improve the accuracy and precision of your hardware, giving you a full free noise 12 bit resolution (4096 steps) in all analog variables.

TUSBA is available in two models:

a) **TUSBA R1**: provides conversion for the 5 analog axes and 13 buttons (10 standard buttons plus the center position of Comms, Dogfight and Speed Brakes switches) and it is not firmware upgradeable.

b) **TUSBA R2**: is a device externally similar to TUSBA R1 but with notable enhancements as more memory and program space, firmware upgradeable and support for the standard axes and buttons and the next extra axes and buttons:

- Three additional axes for throttle control, called “Idle”, “Mil” and “Afterburner”

that allow advanced users to configure the throttle axis with more possibilities, like a “reverse” in Flight Simulator for instance.

- Three DX (DX1, DX6 & DX9) buttons to indicate when switches with more than one position are in neutral position. These buttons can be used to associate their center position with any function of a simulator.
- 16 DX (DX17 to DX32) buttons which can be configured to turn ON or OFF when a specific value is reached in some analog axis.

TUSBA R2 is the object of this User Guide, so hereinafter when you read TUSBA you must understand always TUSBA R2. If you are interested in TUSBA R1 you can see its specific information in the product’s website:

(http://www.realsimulator.com/html/tusba.html),

Beside the above characteristics, the product is supplied with a firmware update tool (DCC) to install new versions of firmware and a GUI application (RS_HID_DEV_TOOL) to configure and calibrate the device. Both tools can be downloaded from the product’s website.

PICTURE

DESCRIPTION

Package content

TUSBA is supplied as a plug and play device with no necessary hardware installation or drivers. The package, showed in the before picture, contains the following components:

1. TUSBA adapter.
2. USB type A extension cable 0,5m.

Technical data

- D-Sub 15HD female socket.
- USB - A male plug.
- Plastic sealed cover.
- 14 bits analog conversion and 12 bit measures full free noise.
- Low pass filter (Glitch Reject).
- 5 analog channels (Throttle, Cursor X and Y, Range and Antenna Elevation).
- 13 DX buttons (10 standard buttons plus the central position of Comms, Dogfight and Speed Brakes switches)
- 3 additional axes: Idle, Mil, Afterburner and 3 associated DX buttons to each

throttle area.

- 16 additional DX buttons linked to analog variables.
- Compliance with Windows XP, W7 or other higher MS operating system, x32 and x64 versions.
- Firmware upgradeable.
- Weight: 60 gr.

PICTURE

DESCRIPTION

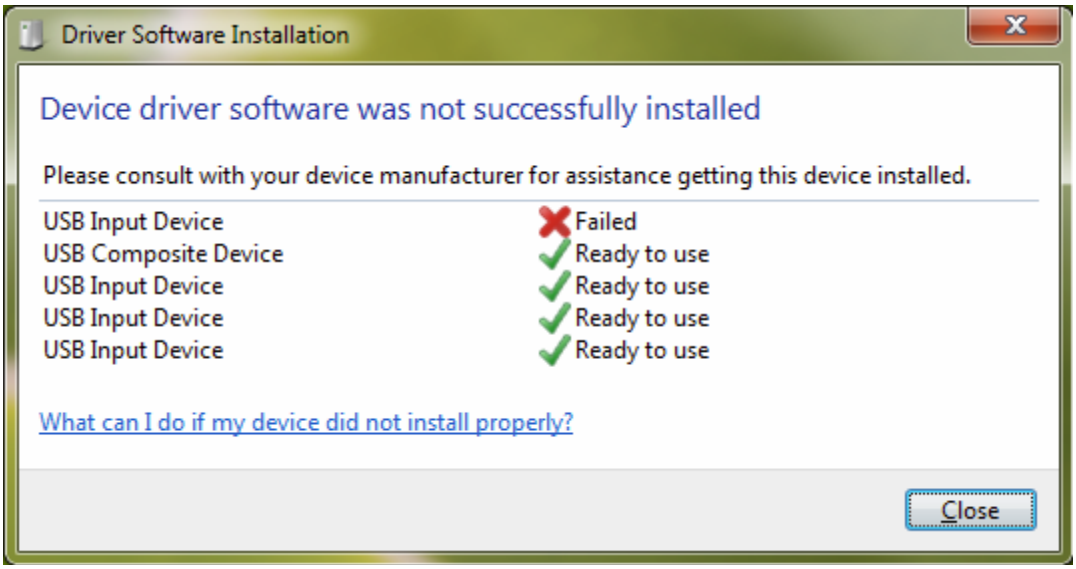
The first time that you connect your TUSBA in your computer, you will take notice about really your TUSBA are several devices living in the same hardware. In fact, inside your TUSBA there are:

- A Boot system, to allow you to update TUSBA with new firmware.
- A HID Game Device, to let you communicate TUSBA directly with MS operating system and let you control your simulator or some other game with DX axes and buttons.
- Two additional comms ports for special comms with the device.

As soon as you connect TUSBA, MS operating system will detect it and will start looking in its data base to install the appropriate drivers for it. As you know, from the FSSB R3 we have avoided custom drivers or additional special system in order not to have any problems in the future with new MS operating systems. Just as you know, every time MS improve its OS, all of us have a headshake with incompatibilities, drivers, etc. so we have learnt from the past, that the best is to use the own legacy MS drivers for comms and this is what we have done in the TUSBA system, use only MS drivers, so when you connect your TUSBA to the computer, MS will look in its data base for the best MS driver for it, in fact its own HID drivers.

The first device to look for drivers is the TUSBA Boot system, and a few seconds after taking comms with MS, the TUSBA Boot will left the system and will be disconnect to allow working the Game device and additional Comms ports. You will see as this USB

input device will be showed in red (have a look to the next picture). And as soon as the Boot goes out a USB Composite Device with 3 USB Input Devices will start looking for drivers.

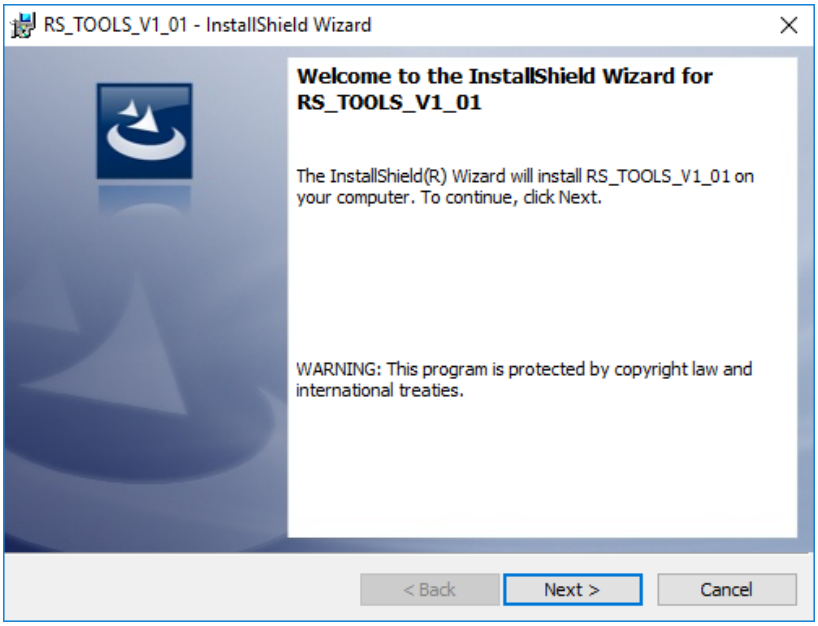


As these HID devices will not go out from your system, after a few second or minutes you will see as your MS OS will found the drivers and install them in your computer and the USB Composite device and the 3 USB Input devices will be tick in green.

This picture is what you normally will have if you are running W7 64bits, perhaps it will look different in other OS, but it will be similar.

About the first USB Input Device, ticked in RED, don't worry it will be connected the next time you upgrade your system with the DCC software. Not necessary now.

PICTURE



DESCRIPTION

The product is supplied with two software tools, a firmware update (DCC) to allow installing new versions of firmware in the device and a GUI application (RS_HID_DEV_TOOL) to configure the device.

Both tools can be downloaded from the download's website inside the same package. The package also includes the RealSimulator device metadata files, latest firmwares and the product User Guide.

To download the tools, please, go to the RealSimulator download's website by clicking the next hyperlink:

http://www.realsimulator.com/html/download.html

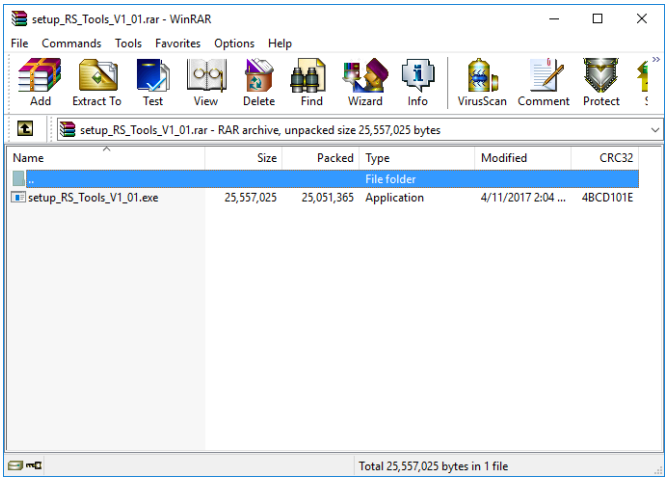
and download the latest version of **RS_TOOLS** and save it where you prefer.

NOTE: If you have a previous version installed in your computer, please, uninstall the older version prior to install the newer one.

System requirements are Windows XP sp3 or above MS operating systems and Microsoft .NET 4.5 or above.

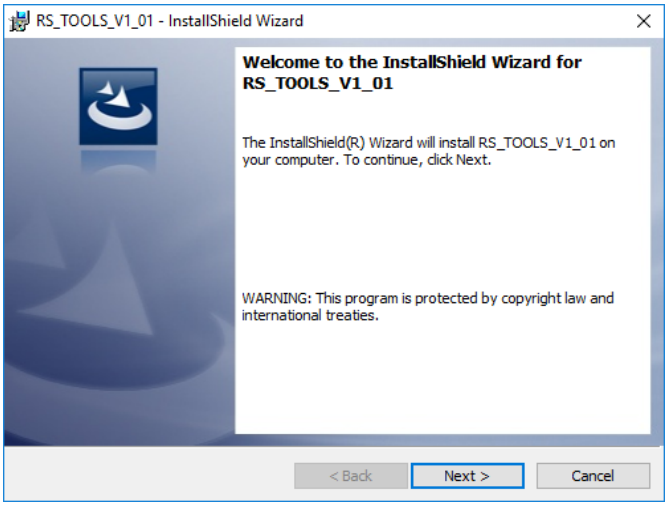
To install it, please, run the downloaded program by double-click on the file icon, at this moment "setup_RS_Tools_V1_01.rar" although the procedure is equal if there is other newer version.

Any case, you should have a window like this one.

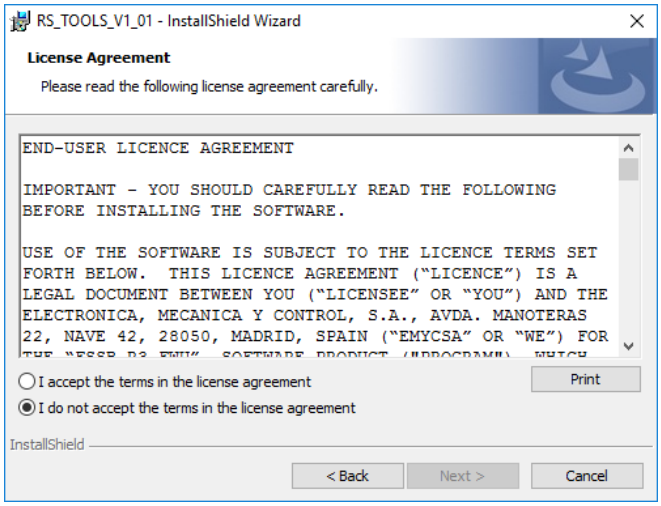


where **setup_RS_Tools_V1_01.exe** is the tools installer. To install it, please, run the file with a double click on the file name.

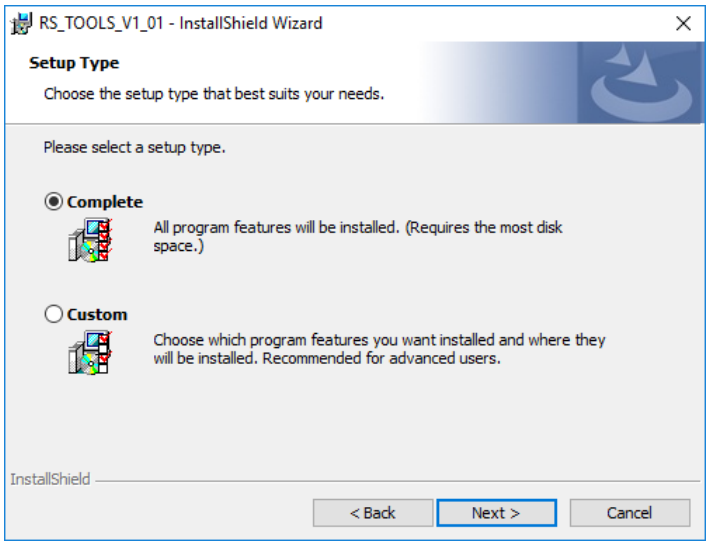
After some seconds extracting and decompressing the package the installation wizard will launch, then select **Next** to continue.



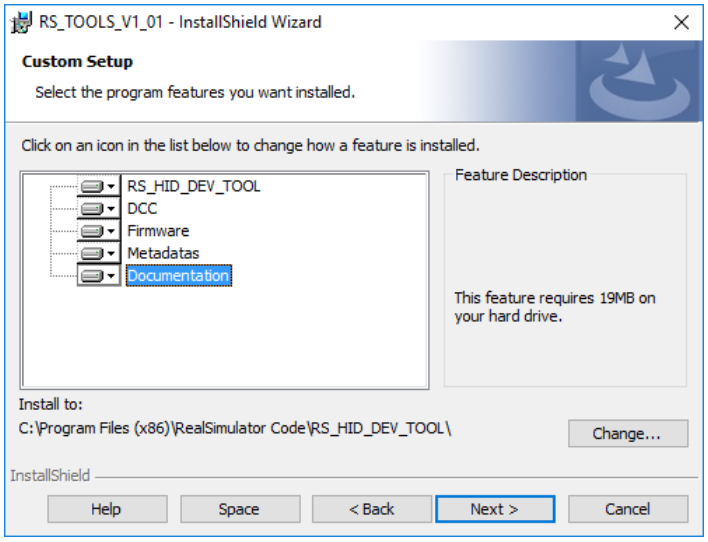
A standard licensing agreement must be accepted before moving on. Choose **I accept the terms of the license agreement** and click **Next**.



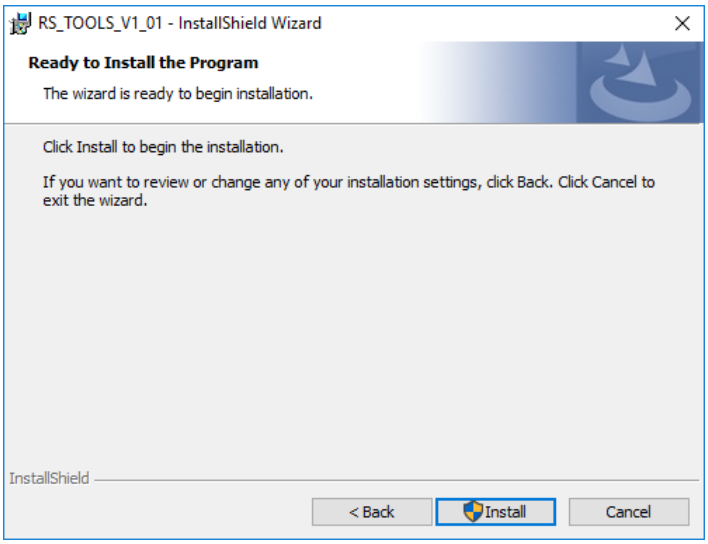
If you want to install all program features (DCC, RS_HID_DEV_TOOL, Metadata files, firmwares and User Guide) select the **Complete** setup type and click **Next** to continue.



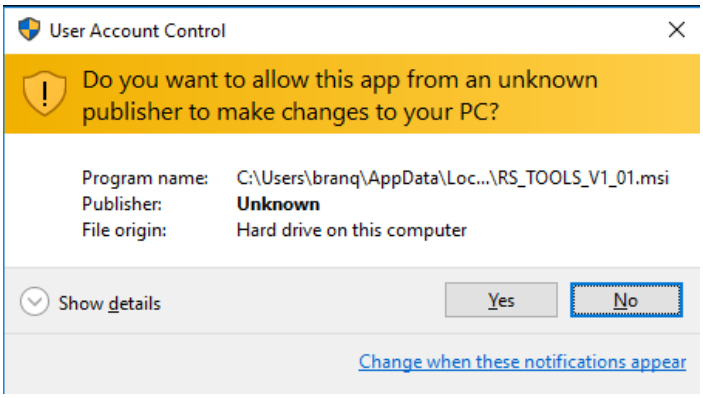
Or select **Custom** if you want to choose the features to install and click **Next** to continue. Select the features to install and click **Next**.



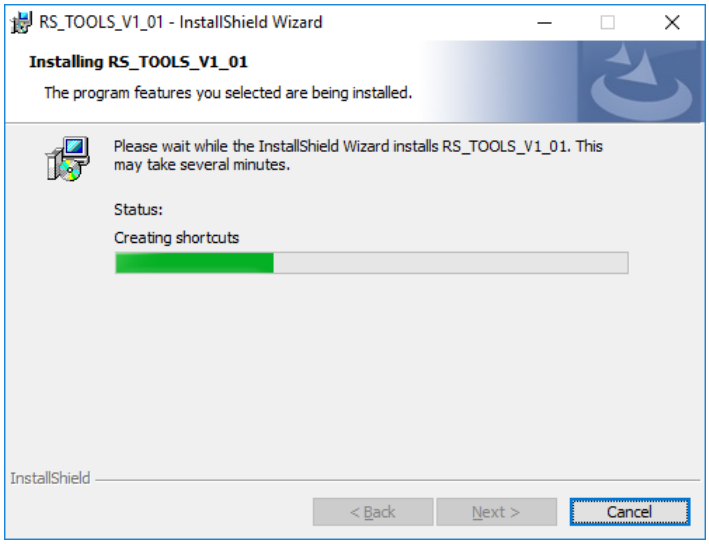
The wizard is now ready to start the installation process. Please, click on **Install**.



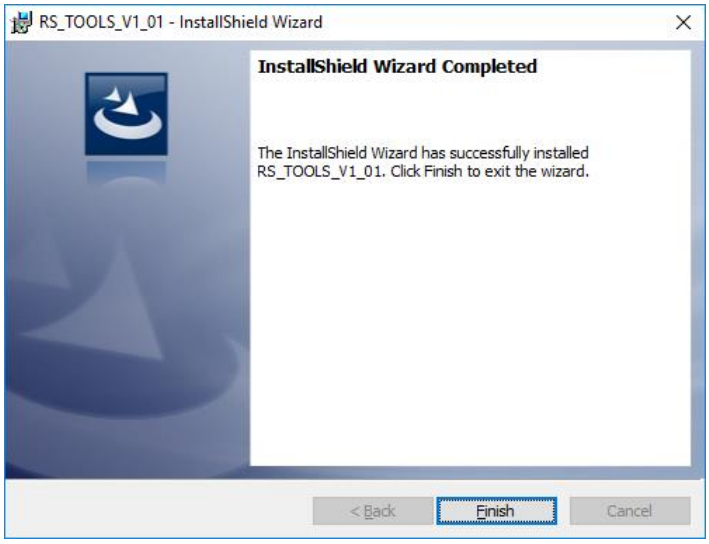
If the User Account Control window appears click the **Yes** button to continue.



The installation of RS_TOOLS_V1_01 may take several minutes to complete.



Wait until the wizard finishes the installation and click **Finish**.

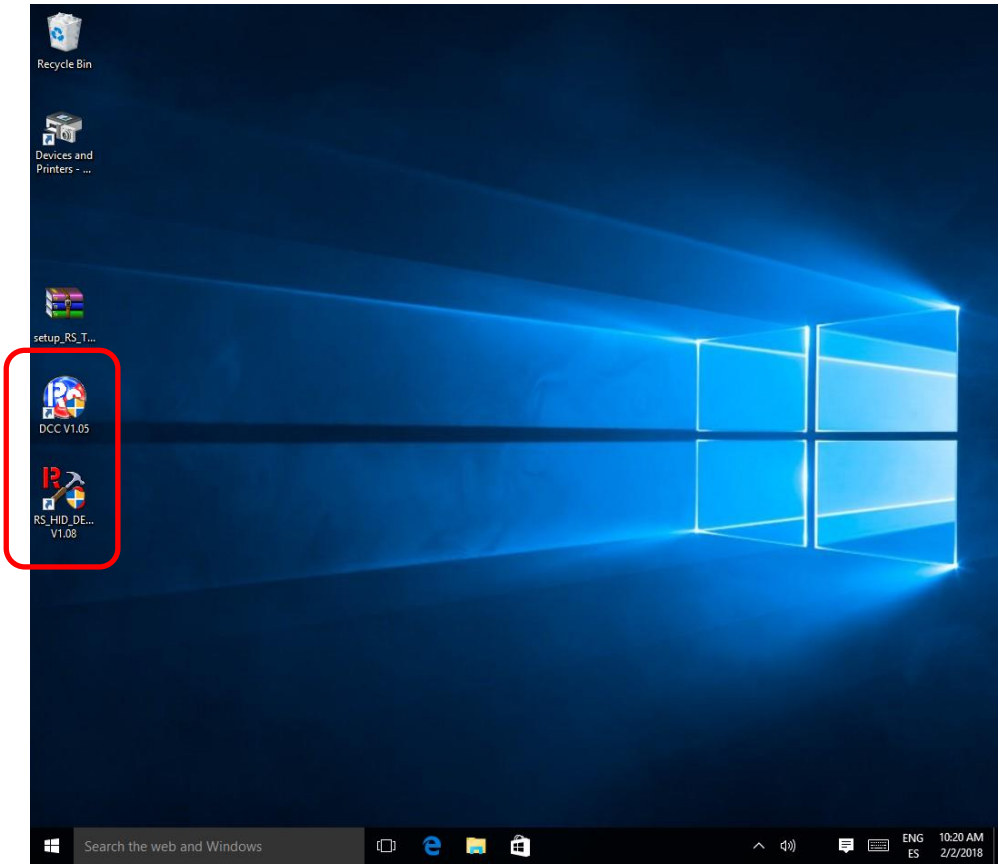


The installation is now finished, if you have connected to the PC your flight controls with the TUSBA kit installed you can see a new device image in the Devices and Printers window.

For this, click in the windows **START** button and select **Devices and Printers**. You should see an icon device like this

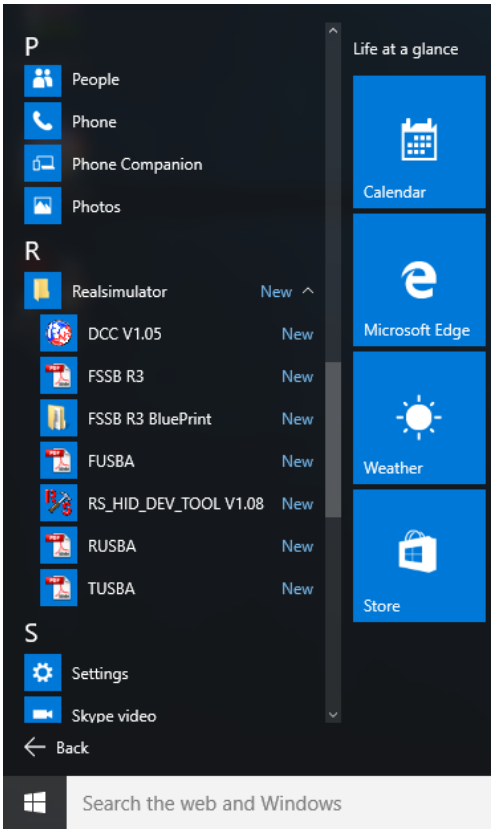
NOTE: sometimes Windows does not update immediately the icon device and you see the standard game device icon, in those occasions press the **F5** key to force windows to update the icon devices cache.

Also, after the installation you will find in your desktop, two new icons: DCC and RS_HID_DEV_TOOL application shortcuts.



Finally, if you press the windows **START** button and look the **All App** section in the R letter you will find in the Realsimulator folder shortcuts to the DCC and RS_HID_DEV_TOOL programs, RealSimulator devices User Guide and the FSSB R3

BluePrint.



As general information, here it is the default directories where the installer saves programs:

C:\Program Files (x86)\RealSimulator Code\DCC
C:\Program Files (x86)\RealSimulator Code\RS_HID_DEV_TOOL

And, the User Guide, Metadata, Firmware and BluePrint files on:

%APPDATA%\Realsimulator Data\

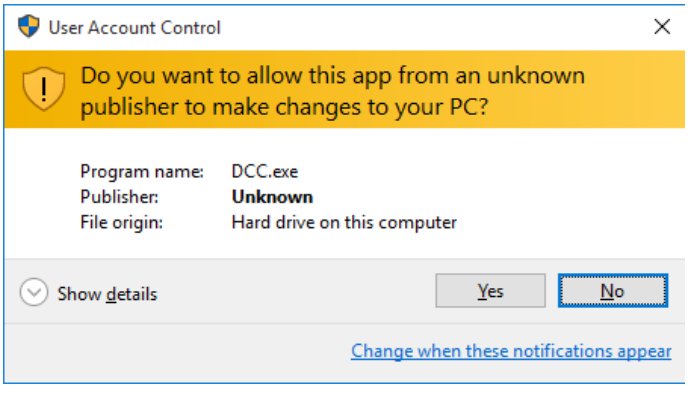
PICTURE

DESCRIPTION

Usually you will receive your RealSimulator device with the last firmware version installed, so it will not be necessary to use DCC to update your device immediately after its reception.

If you check the product's website periodically you could find new versions with enhancements and issues fixed, so you will need to use the DCC program.

To start the DCC program, launch by double click the DCC desktop icon or click in the windows **START** button and select **All Program > Realsimulator > DCC > Launch DCC.exe.** If the User Account Control window appears click **YES** to continue.

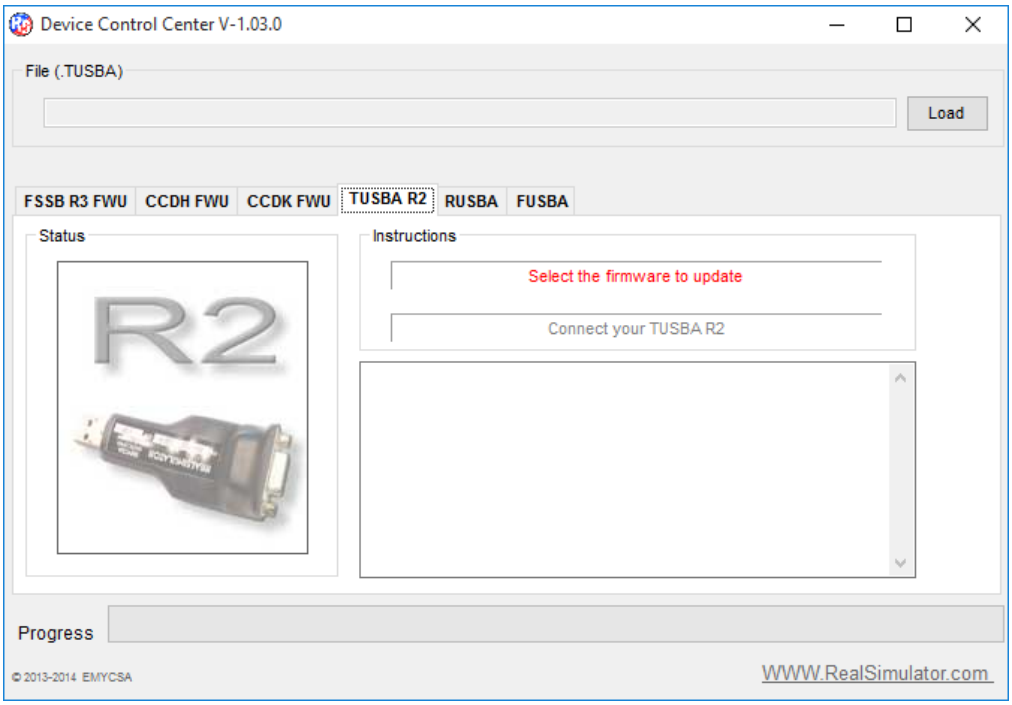


Select the tab labelled as **TUSBA** and follow the instructions given in the groupbox

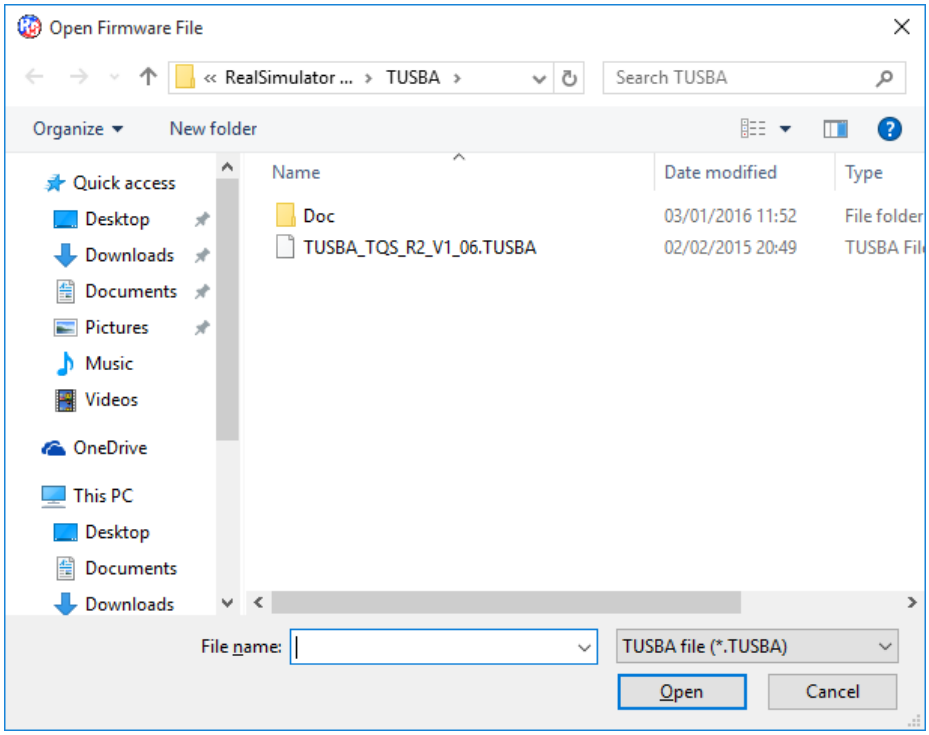
Instructions to update the device.

Unplug the device from USB extension cable supplied or computer port.

As you can see in the next picture, **Status** groupbox shows a light device image, it is normal, this image will only be in normal colour when the device is running the bootloader program, in other cases, with the device unplugged or in normal operation the image will be light.



At first, the **Select the firmware to update** message will be blinking in red, so click the **Load** button to open the Open Firmware File window to select the new firmware to install, select the desire file clicking the filename and click the **Open** button to close the window.



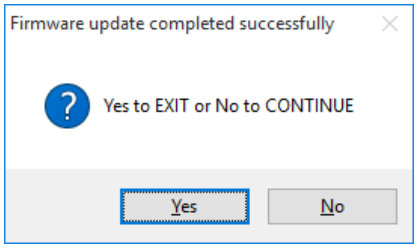
Second, with the previous message in green, the following message **Connect your TUSBA R2** will blink in red.

To start the updating we must plug in the previously unplugged device, which will launch the bootloader for some seconds and DCC program will start the communications with the device sending the new firmware. During this data transference we will be able to see the progress in the Progress bar and the status image in normal colour.

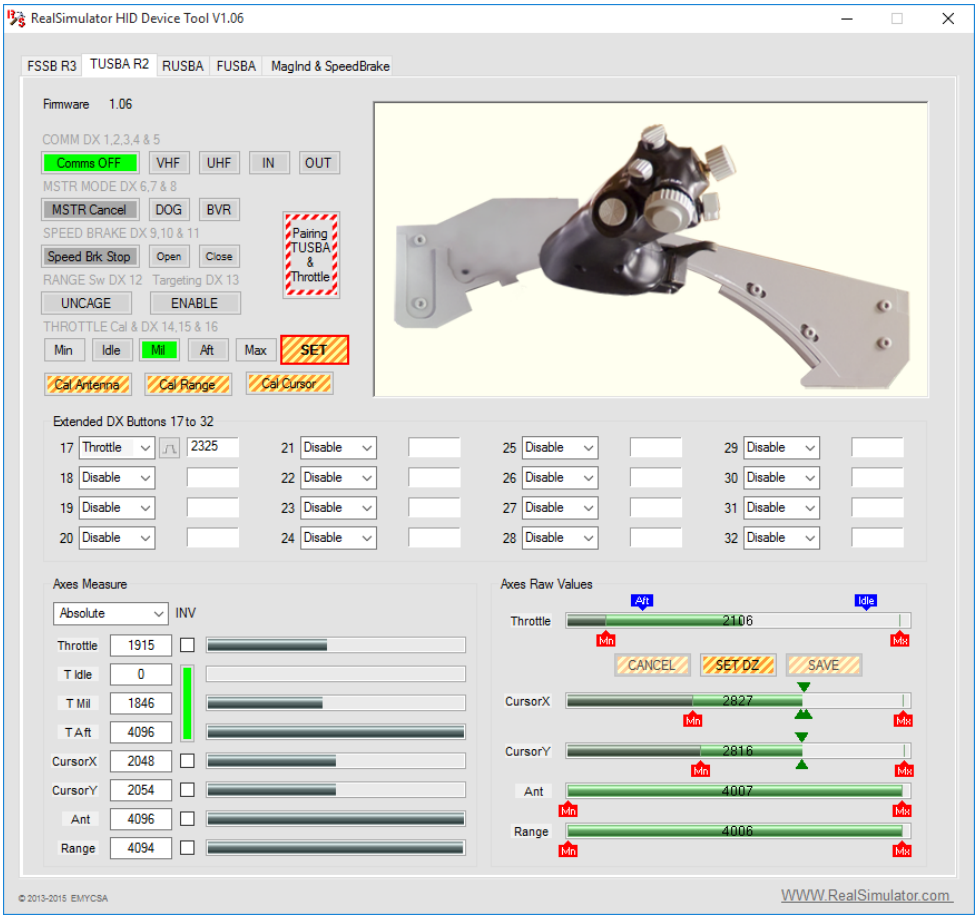
Finally, when the update finishes the device will exit from the bootloader program and will run the new firmware.

DCC program will show a new window to confirm the firmware update completed successfully and it will ask you to continue with other device or exit.

Click **Yes** to exit.



PICTURE



DESCRIPTION

TUSBA, like all devices manufactured by RealSimulator that must be connected with customer's hardware, is shipped without configuring, it is necessary to do this labour by himself when receives the device.

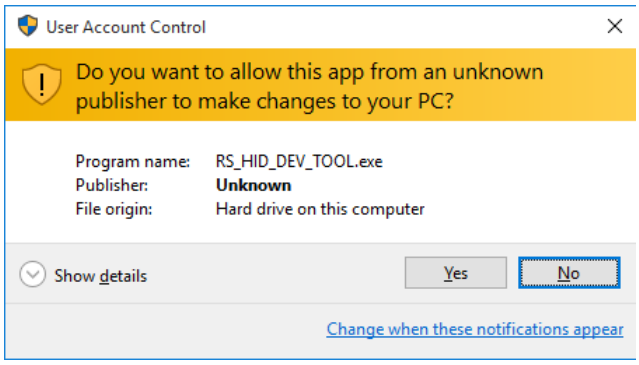
RealSimulator has developed a tool called **RS_HID_DEV_TOOL** to facilitate this action; thereby you will be able to customize and adjust easily the different options offered by the device and get the maximum performance from your hardware.

TUSBA gives functionality for the following axes and buttons:

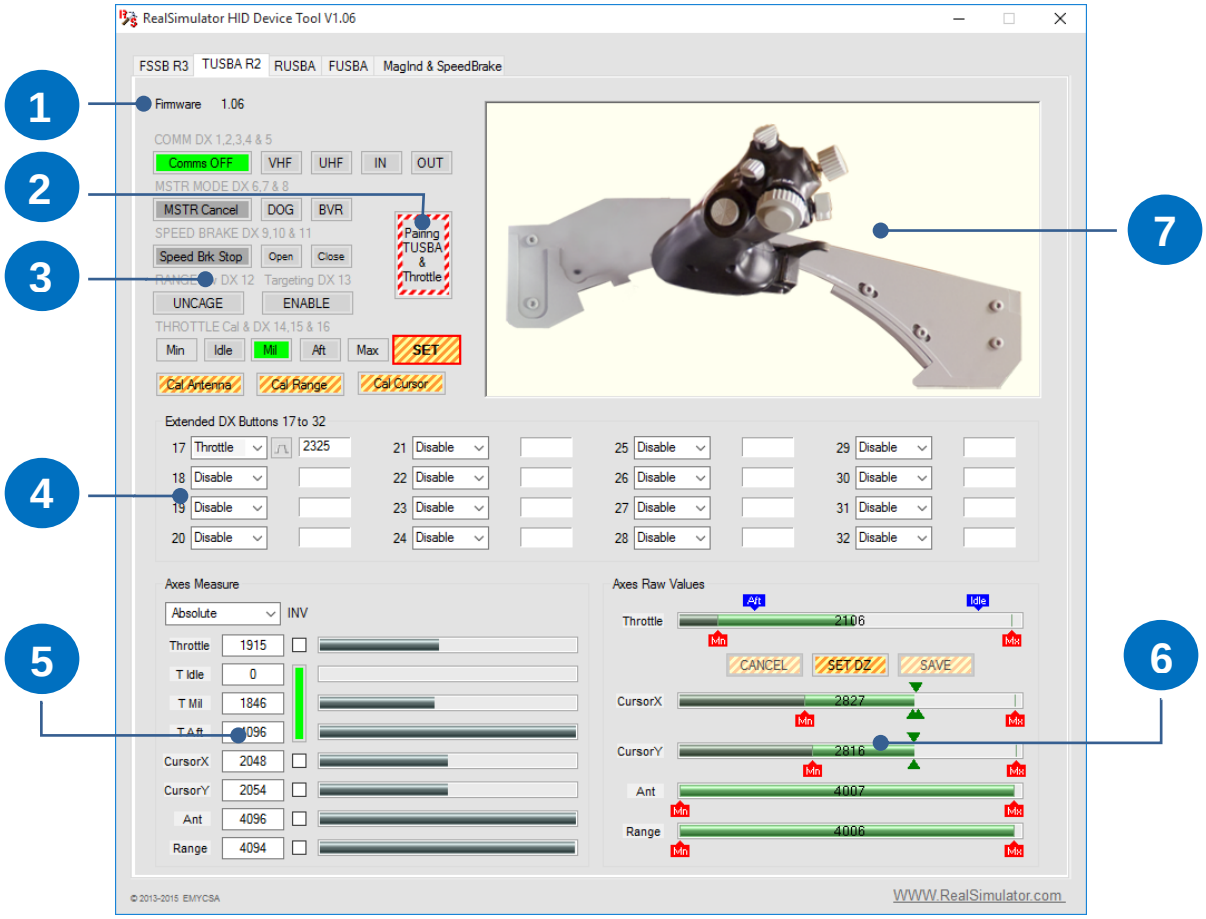
- Throttle, Cursor X and Y, Range and Antenna Elevation axes.
- Three additional axes associated to throttle: Idle, Mil and Afterburner.
- 10 standard buttons of throttle and the center position of Comms, Dogfight and Speed Brakes switches.
- 3 buttons associated to the throttle position.
- 16 additional buttons linked to analog variables.

Following we will explain you how to operate with the RS_HID_DEV_TOOL program and each step required to configure your TUSBA. In order to do this, first we will connect our TUSBA to the PC, directly to an USB port or through the USB expansion cable supplied.

To start RS_HID_DEV_TOOL program, launch by double click the RS_HID_DEV_TOOL desktop icon or click in the Windows **START** button and select **All Program > Realsimulator > RS_HID_DEV_TOOL > Launch RS_HID_DEV_TOOL.exe.** If the User Account Control window appears, click **YES** to continue.



Select the tab labelled **TUSBA R2** and you should see a window like this,



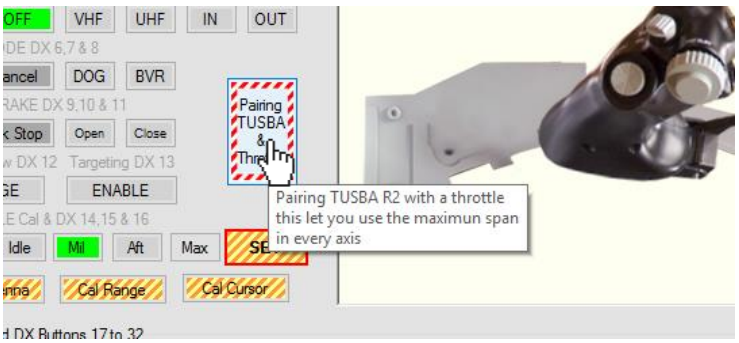
where we have identified with numbers the different information and configuration areas.

1. Area where to see the firmware version installed in the device. If a message of "**Firmware non supported**" is showed you need to update your firmware because you have installed an old version and not full

actions will be supported.

2. Pairing button. Below you will find a complete explanation about its operation.
3. Area with buttons to calibrate individually the analog axes and the status of the DirectX buttons. Buttons are grouped by functions and are identified by its name and DX number assigned.
4. Area where to configure and see the status of the DirectX buttons linked to analog variables.
5. Groupbox with the DirectX information for the analog axes. Here we can see the axes measures as graphical information in progress bars and text box with numerical information and check box to invert the axes. The information showed in this groupbox for the different axes is the same as we can see in the Microsoft Game Controllers window.
6. Groupbox with the Raw information for the analog axes. Here we can see the measure of each real axis numerically and graphically in a progress bar, red icons for the maximum and minimum values of each axis, green icons for the dead zone of Cursor X and Y axes, blue icons for the throttle Idle and Afterburner positions and finally three buttons to make manually the dead zone adjust. Additionally from V1.09 the user can see and adjust the value of each icon manually clicking on the icon and writing the desired value manually (see section **Manually configure Raw Axes**).
7. Animated area where the program shows interactively the throttle position and actions of buttons and switches.

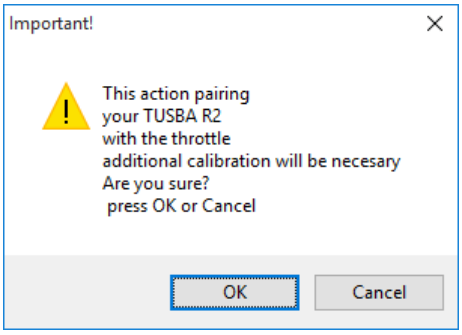
In general, every button or checkbox in the window has associated a small pop-up box (tooltip) with basic information about it. This information will appear when the mouse pointer is over the button or checkbox.



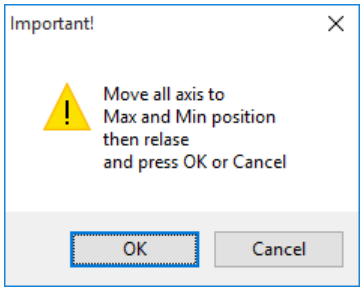
Pairing

The process of adjust and calibration of throttle with TUSBA starts with the Pairing action. With this action TUSBA adjusts individually the amplifiers of each analog axis to obtain the maximum span and so to make better use of 12 bits analog converter. For this, TUSBA needs to know the maximum and minimum values of the analog measure of each axis, so as we will see after, we will need to move the axes to the extreme positions to achieve these values.

To start the pairing process, click the **Pairing TUSBA & Throttle** button, and click **OK** in the new window to continue.



A new window will appear to inform us about **moving all analog axes to maximum and minimum positions** as comment above. It is very important to make this phase well for a future correct operation; we suggest maintaining the end positions for some seconds to guarantee TUSBA identifies correctly end positions. To finish, click "**OK**".



NOTE: It is absolutely necessary to make the pairing action at the first time when receive TUSBA, but it is also advisable after a firmware update.

Axes calibration overview

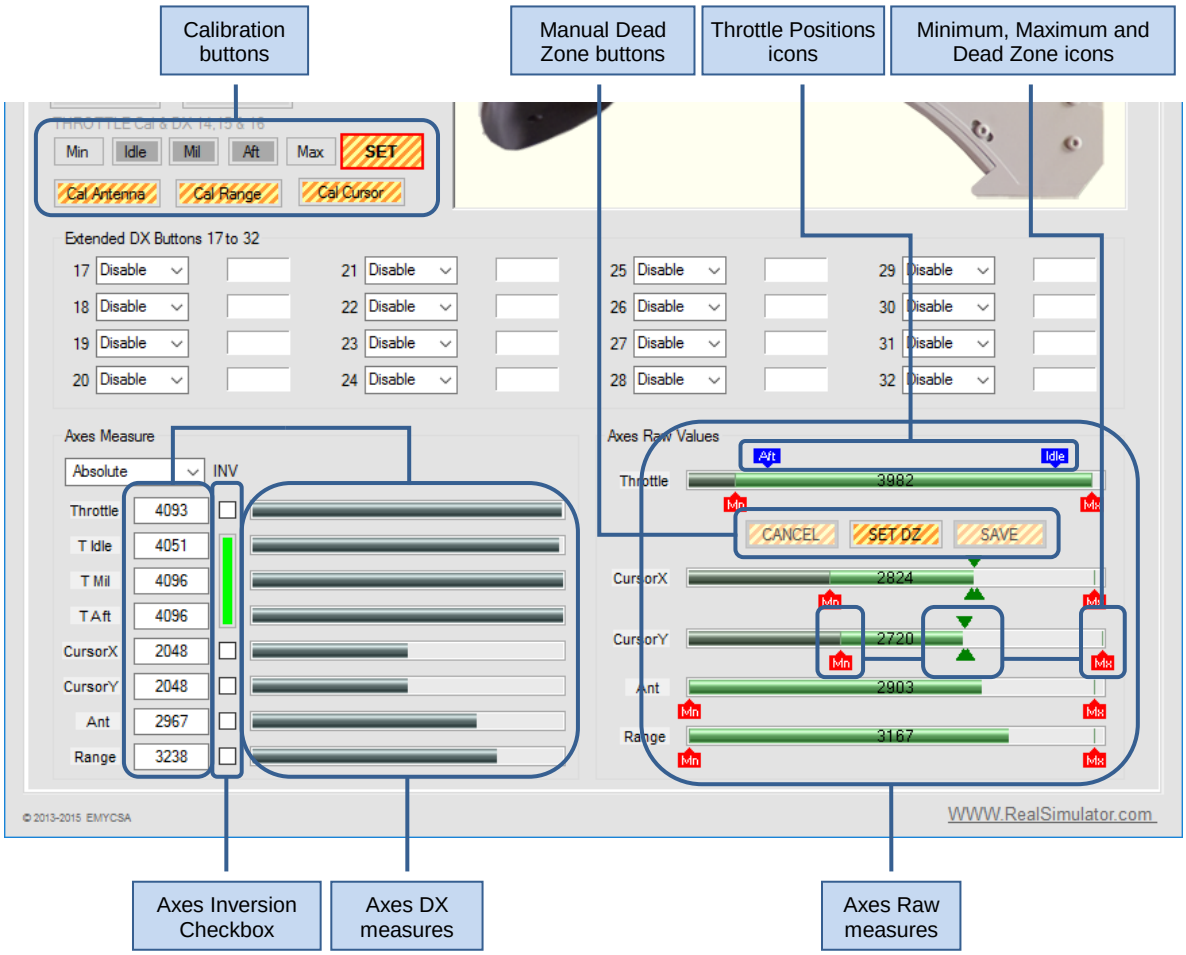
As the pairing window informed us before, after the pairing is necessary to calibrate each analog axis.

With this calibration we will inform Windows about the maximum, minimum and central position of each axis. This procedure is similar to the Game Controllers properties calibration wizard of Windows with the following differences:

- With the RS_HID_DEV_TOOL we can calibrate individually axes, only the axis that we need.
- If the axis has central position, like the Cursor axes, RS_HID_DEV_TOOL allows us the possibility of adjusting a dead zone area (automatically and manually).
- Invert individually axes to adapt the hardware to the simulation program requirements.
- See the Raw and DX values graphically and numerically.

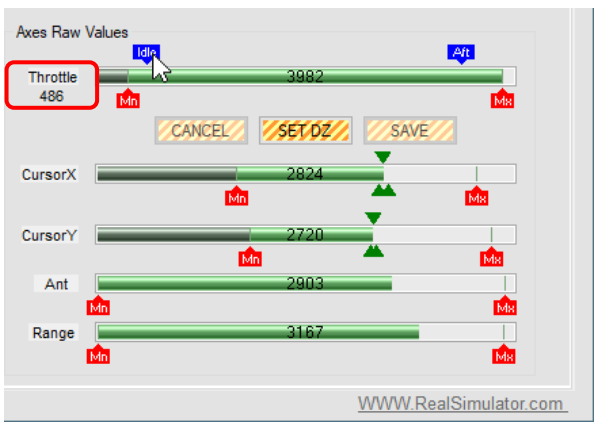
So, **we only suggest using this tool to configure TUSBA and calibrate the throttle**.

In the next image, you can see identified the different informative areas related with the axes calibration.

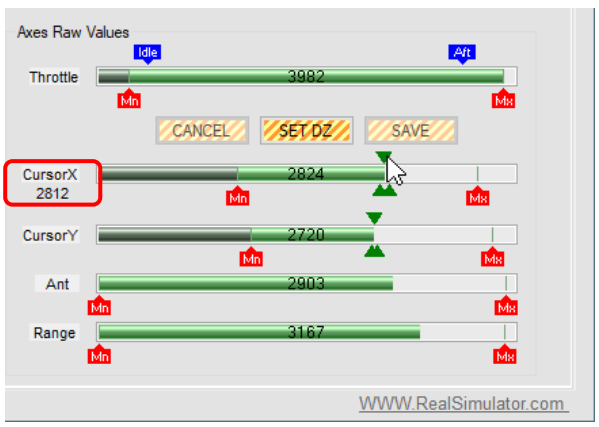


Axes Raw Values groupbox shows the internal measures of each TUSBA analog axis, numerically and graphically through a green progress bar with the numeric value in the centre. Each axis has associated two red icons to identify the maximum and minimum positions; if the axis has central position, it has associated a dead zone identified for three green icons and finally the special case of throttle with two blue icons to identify the Idle and Afterburner positions.

Values of associated icons (Maximum, Minimum, Dead Zone, Idle and Afterburner) can be visualized putting the mouse over the icon and modified by the user by double-click over the icon. For an explanation about how to do it, please, see the section **Manually configure Raw Axes**.



Throttle Idle value

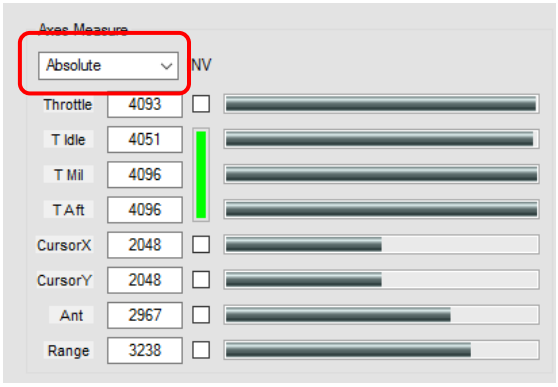


CursorX Centre Dead Zone value

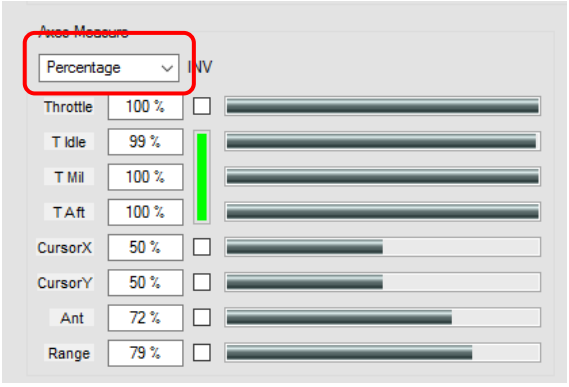
If you move one axis of Cursor, you can check the information showed is “fully raw”,

because the dead zone area is not applied to the measure showed. To see the "normally raw" values that TUSBA sends Windows to generate the DX values is necessary to open the Game Controllers properties window of device and enter in the Calibration wizard. We have made the decision to show the "fully raw" measures because we consider this is more interesting than the other one and allow us to show graphically the full information to the user.

Axes Measure groupbox shows the Direct X information. This information is showed numerically on text boxes and graphically in progress bars. The numeric information can be showed in **Absolute** or **Percentage** format. To change it, select the preferred option in the combo box.



Absolute format



Percentage format

TUSBA, apart from the five real analog axes (Throttle, Cursor X and Y, Antenna and Range), gives functionality for three extra axes associated to the throttle and identified as Throttle Idle, Throttle Military and Throttle Afterburner. These three extra axes are generated from the real throttle axis and their courses are associated to the three zones that Idle and Afterburner icons/buttons delimit in it.

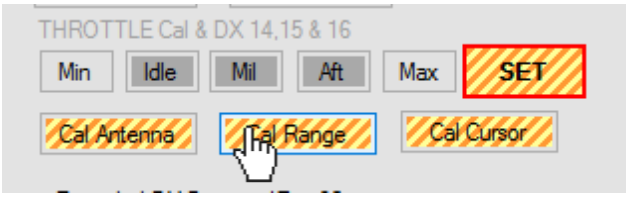
The axes calibration procedure is very easy and is guided with instructions on the screen; however, we are going to show you step by step how to do it.

The axes calibration is grouped in three paragraphs:

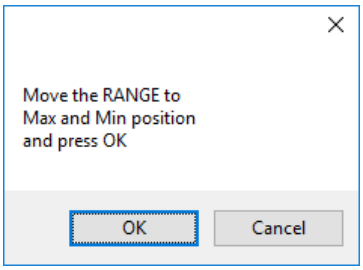
- Calibration of axes with no central position for the **Range** and **Antenna** axes.
- **Throttle** calibration, including the throttle **Idle**, **Mil** and **Afterburner** axes.
- Calibration of axes with central position for the **Cursor X** and **Y** axes.

Calibration of an axis with no central position

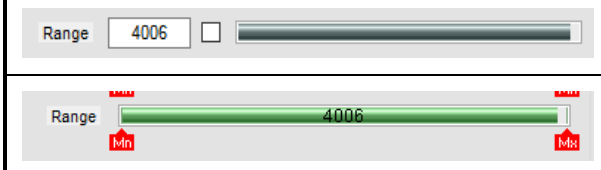
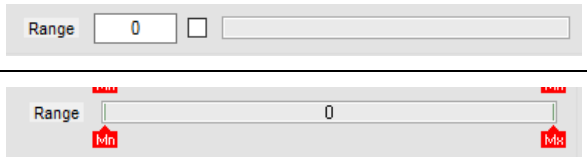
We are going to show you how to calibrate a standard axis with no central position, for example, the Range. For this, click the **Cal** button associated to the axis.



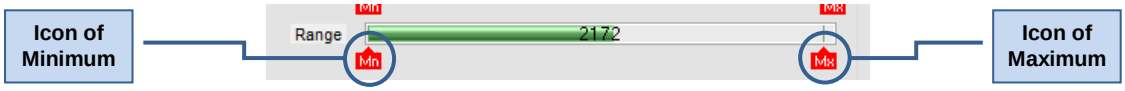
And a new small window will appear with instructions to make the axis calibration.



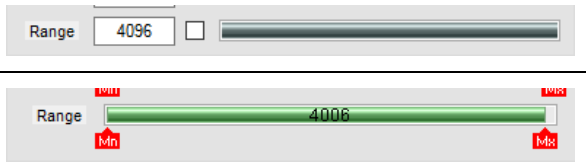
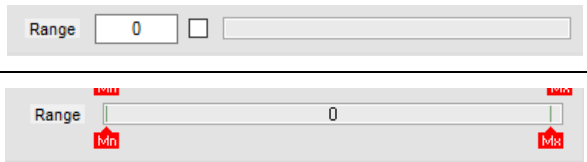
Following the instructions, turn the Range rotary to maximum and minimum positions, and finally click **OK**.

Maximum Position	Minimum Position
	

After closing the window, RS_HID_DEV_TOOL will save the new calibration values for Windows and will relocate the red icons of maximum and minimum to the new positions, as we can see here.



Now, after the calibration there is a direct correlation between the maximum and minimum positions of our hardware, which are showed with red icons, and the maximum and minimum values of DX, as we can see in this table.

Maximum Position	Minimum Position
	

In order to finish the axis calibration/configuration, we have to know that RS_HID_DEV_TOOL gives us the possibility of inverting the axis. To do it, click in the check box associated to axis and automatically the measure and the maximum and minimum positions will be inverted, as we can see in the next image.

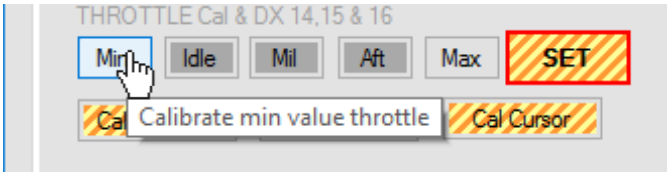
Before

After

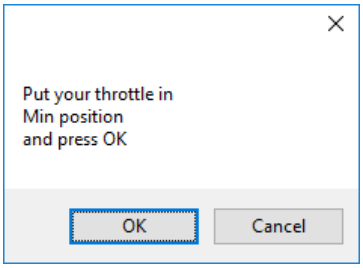
Throttle calibration

The throttle is an axis with no central position, similar to standard axes mentioned previously but as it has associated others axes (Idle, Mil and After) we have preferred to do it in a special section.

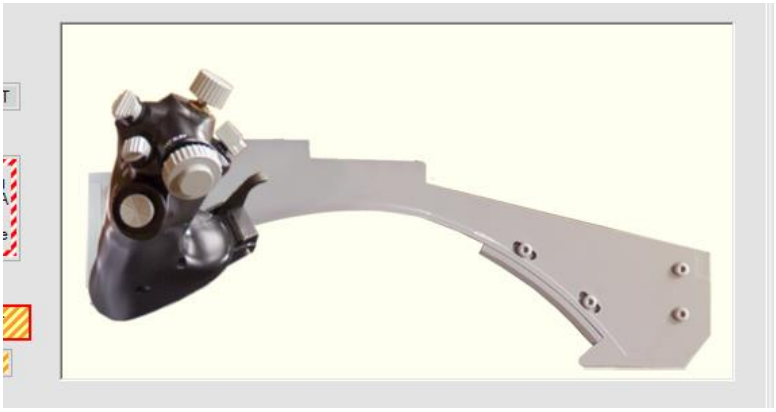
In this case we do not have a dedicated button to calibrate and get the minimum and maximum values of axis, here we have two dedicated buttons, one for minimum (Min) and another for maximum (**Max)**. So, we will start clicking on the **Min** button.



And a new small window will appear with instructions to set the **Min** position.



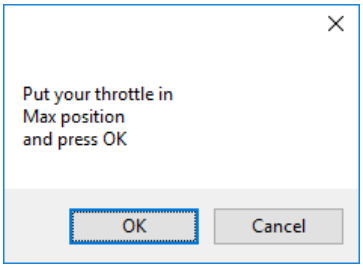
Following the instructions, move your throttle backward until the extreme position and click **OK**.



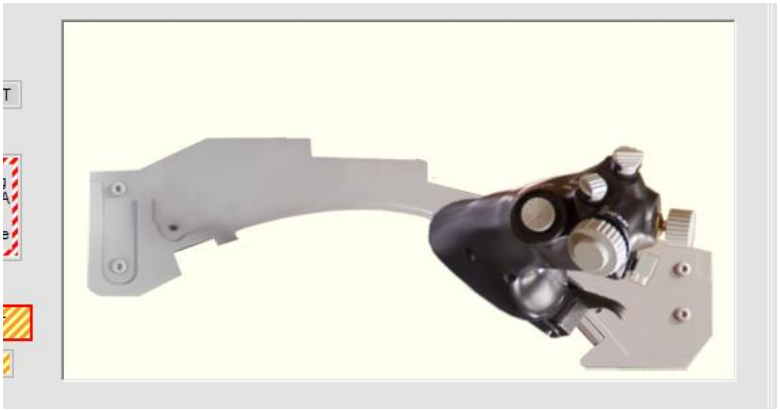
As we can see in the previous images with the **Min** button we set the “Mx icon”. This looks like incongruent, but we have to understand that Raw values and its red icons have relation with the electrical signal and the Cougar throttle hardware gives maximum signal in the minimal position of throttle, and positions and buttons of **Max** and **Min** have relation with the physical position of the grip and with the power. This does not have any implications in the TUSBA operability, but I have considered important to clarify these concepts.

Now, we will continue the calibration pressing the **Max** button to set the “Mn icon”.

And a new small window will appear with instructions to set the **Max** position.

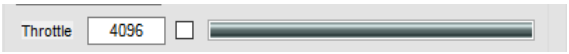
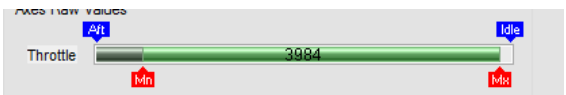
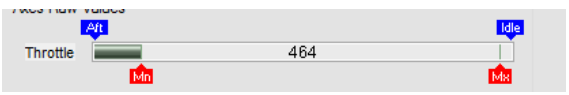


Following the instructions, move your throttle forward until the extreme position and click **OK**.



And, as we can see in the before images we have set the “Mn icon” for the maximum position.

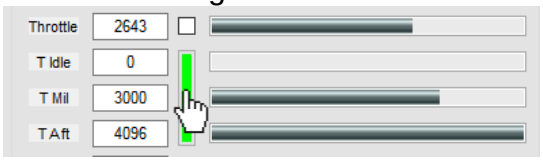
Windows already has all information it need, the maximum and minimum values for the maximum and minimum positions and as we can see in this table (for the moment, ignore the blue icons) there is a direct correlation between the maximum and minimum positions of our hardware, showed with the red icons, and the maximum and minimum values of DX.

Throttle “Min” Position	Throttle “Max” Position
	
	
	

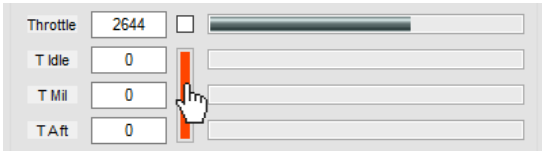
Now we will see how to calibrate the three auxiliary axes of throttle: Idle, Mil and Afterburner.

Before to start, we must to know that there is a button to enable/disable the auxiliary throttle axes; you can change the status by click on it. If the button is in green the auxiliary axes are enables and if it is in red, the axes are disables and the measures are “0”.

This functionality allows, for example, avoiding problems when we are configuring a game and the assignation of axes is automatic by the movement of axes.



Throttle auxiliary axes enable

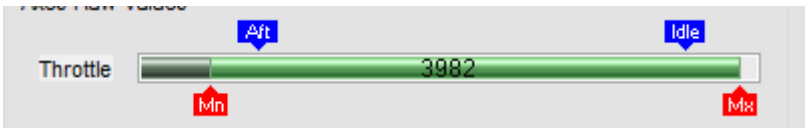


Throttle auxiliary axes disable

As these auxiliary axes are not real axes, we cannot make a real calibration of them but we need to define the course of each axis.

In this case, we will assign two points, concretely the **Idle** and **Afterburner** positions, so we will have defined three areas over the real throttle course that are the three axes:

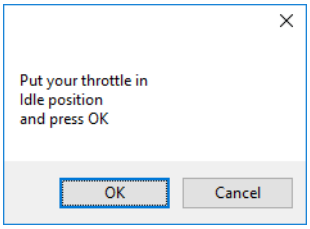
- **T. Idle**: area defined by red “Mx icon” and the blue “Idle icon”.
- **T. Mil**: area defined by blue “Idle icon” and the blue “Aft icon”.
- **T. Afterburner**: area defined by blue “Aft icon” and the red “Mn icon”.



These three areas are symbolically represented in the **Idle**, **Mil** and **Aft** buttons (DX buttons 14, 15 and 16 respectively), so if they are enabled, we will have a button lighted in green showing in which area is the throttle. You can find more detailed information about these buttons in the DX buttons paragraph below.

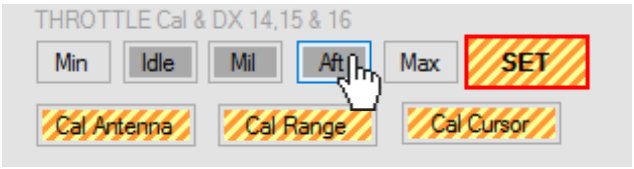
In order to start the calibration of auxiliary axes, we are going to define the **Idle** position, for that, click the **Idle** button with the mouse left, the mouse right has other function assigned as we will see later.

And a new small window will appear with instructions to set the **Idle** position.

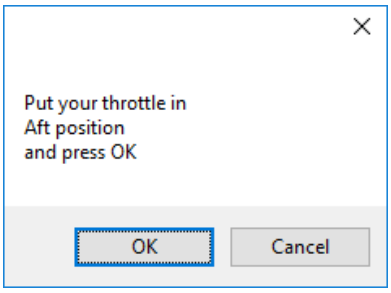


Following the instructions, put your throttle in the Idle position, for that, move your throttle backward until the extreme position and next, move the throttle forward until the first hardware detent and click **OK**.

Now, to define the **Afterburner** position, click the **Aft** button with the mouse left, as the Idle button, the mouse right has other function assigned.

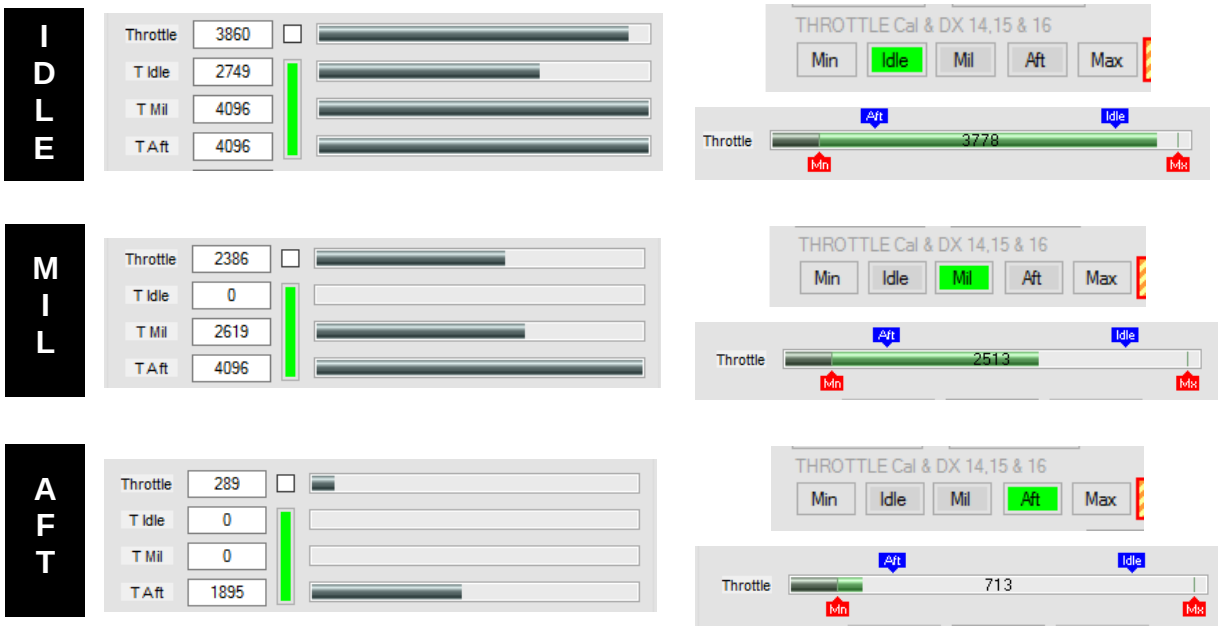


And a new small window will appear with instructions to set the **Aft** position.



Following the instructions, put your throttle in the **Aft** position, for this, move your throttle forward until the next hardware detent and click **OK**.

To finish, verify the configuration/calibration of auxiliary axes is well done, as we can see in the next table.

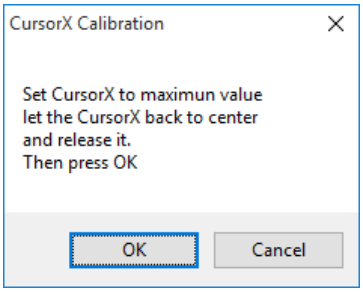


Calibration of an axis with central position

Now we are going to show you how to calibrate one axis with central position, the Cursor, which has two axes, X and Y. As before, it will be necessary to get the maximum and minimum values of each axis, but as the axis also has a central position, it is necessary to detect this position and the dead zone around this central position. For that, the calibration in this type of axes is made in two phases.

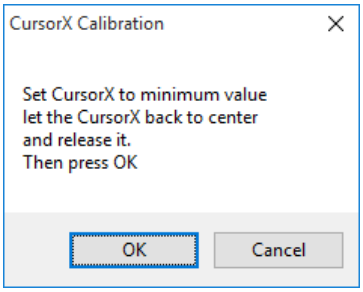
To start the calibration, click the **Cal** button associated to the axis.

A new small window will appear with instructions for the first phase of the X axis calibration.



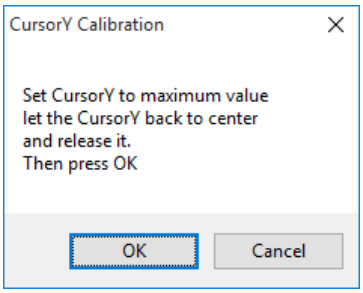
Following the instructions, move the cursor to the left to achieve the maximum value in the raw measure, let the cursor back to centre and release it. Click **OK** to continue.

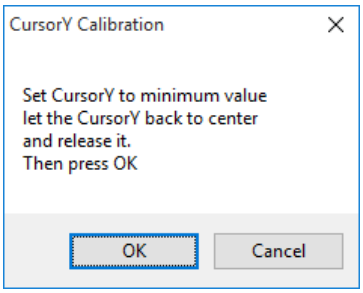
A new second window will appear with instructions for the second phase of X axis.



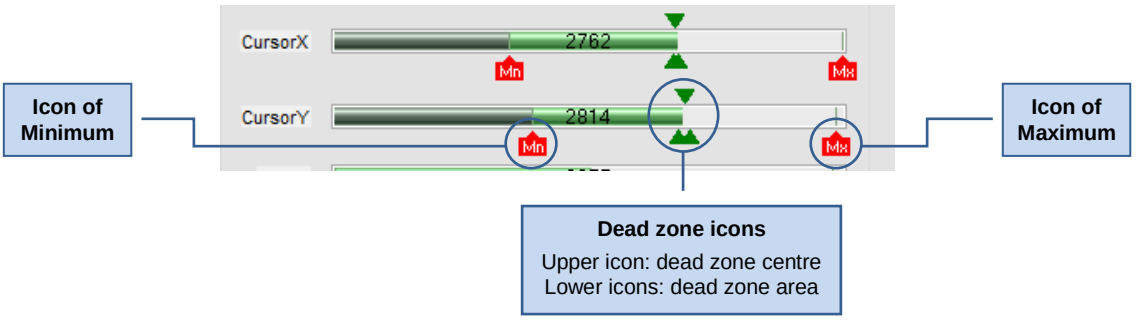
Now move the stick to the right to achieve the minimum value in the raw measure, let the cursor back to centre and release it. Click **OK** to continue.

Now we will repeat the process with the Y axis, moving the cursor forward to achieve the maximum value and backward to the achieve de minimum.

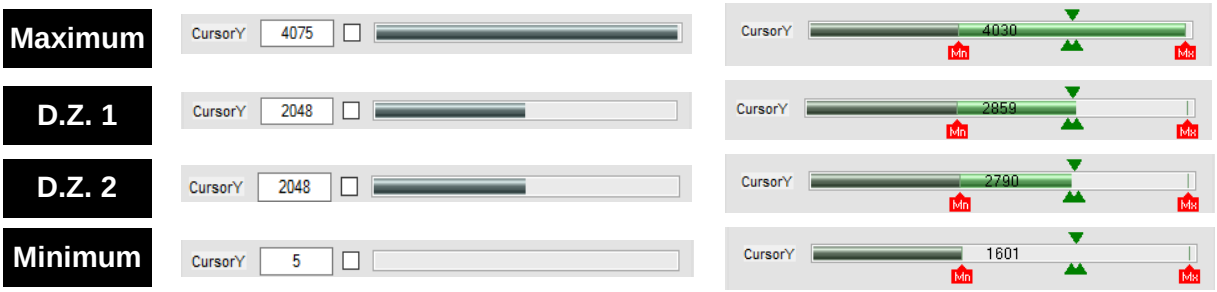




After closing the last window, RS_HID_DEV_TOOL will save the new calibration values for Windows and the automatic calculated dead zone. It will also relocate the red icons of maximum and minimum positions and the green icons of dead zone area, as we can see here.



Now, with the calibration finished there is a direct correlation between the maximum and minimum positions of our hardware showed with the red icons and the maximum and minimum DX values, and a dead zone area delimited for the two lower green icons, where DX measure of axis does not change, although the raw measure changes inside them, as we can observe in the next example with the Y axis values.

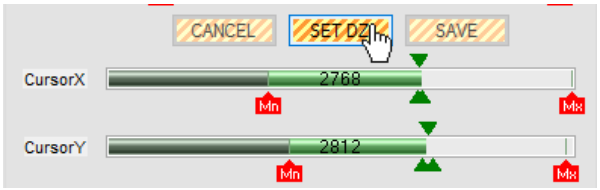


As we see in the calibration of Cursor X and Y axes, RS_HID_DEV_TOOL has calculated the dead zone values in function of only one movement of cursor in each axis. Normally the calculated values are valid to operate, but sometimes if our

hardware is very used and has much play in the centre, then the dead zones calculated are not enough and we could need to adjust them manually.

If when the cursor comes back to the central position their X or Y positions are out the dead zone area, we can be absolutely sure it is our case and we will need to repeat several times the movement and release of cursor, and get visually the new desired dead zone areas to adjust them manually how we explain below.

To manually adjust the dead zone, click the **SET DZ** button

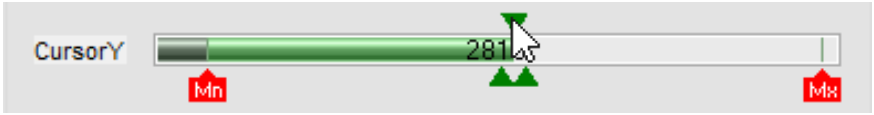


Automatically, if it is possible, the program spans the axes with dead zone to achieve more precision in the adjust process and enable the green icons movement. When the axes span occurs, the program adjusts automatically the progress bar width between the maximum and minimum positions.

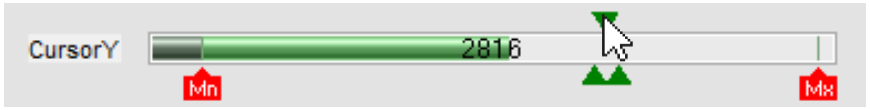
To adjust the dead zone, the program allows us two possible actions:

- To displace the dead zone area.
- To increase or decrease the dead zone.

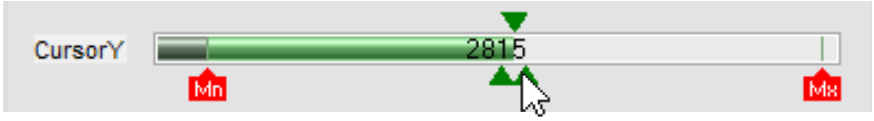
To displace the dead zone area, click and hold with the left button of mouse on the upper triangle and move it to the desired position, and release the button to finish. As you can see you displace the three triangles like a block.



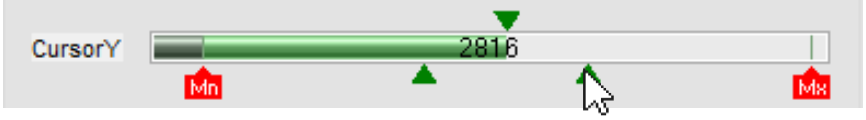
Clicking the upper green triangle and without release the mouse button you can displace the area where you want



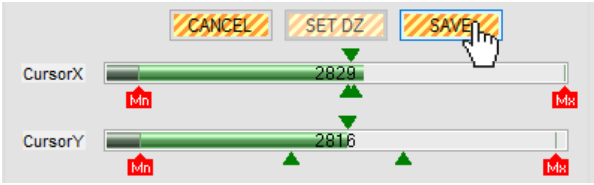
To increase or decrease the dead zone area, click and hold with the left button of mouse on either of lower triangles and move it to the desired position and release the button to finish. Both lower triangles will move symmetrically with respect to the upper triangle.



Clicking on either of lower green triangles and without release the mouse button you can increase or decrease the dead zone area



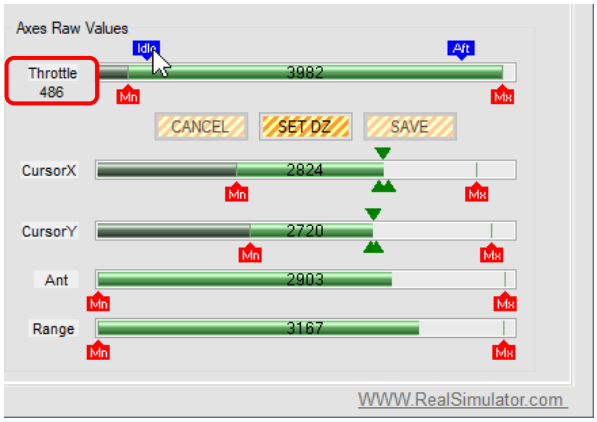
When the dead zone adjust is finished, click the **SAVE** button to save the new adjust configuration or click **CANCEL** to restore the previous adjust.



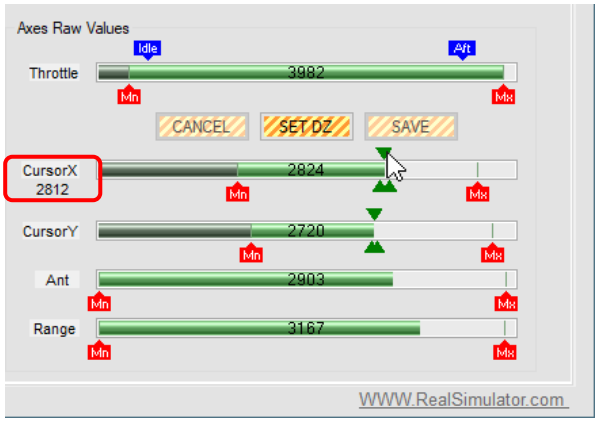
Manually configure Raw Axes

From v1.09 of RealSimulator HID Device Tool is possible see and adjust manually each parameter associated to the raw axes progress bars, as Max, Min, Idle, Aft and Dead Zones.

To see a parameter value you must put the mouse over the parameter icon and the measure will be showed for 4 second below the raw axis name. After this time the measure value disappears and will be necessary to move the mouse and put again over the icon to see it again.

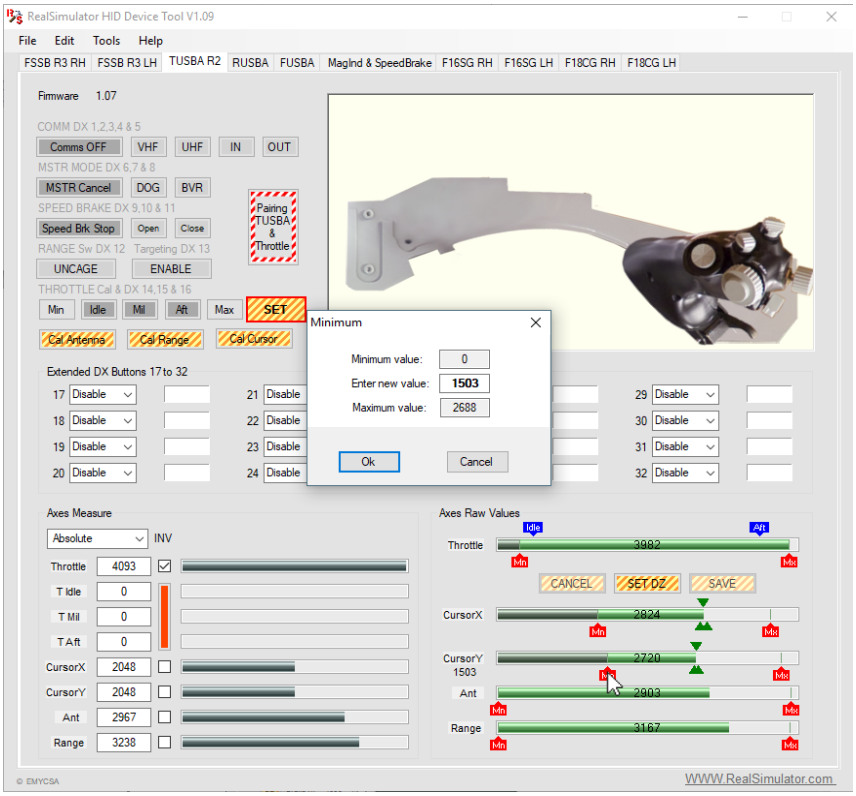


Throttle Idle value

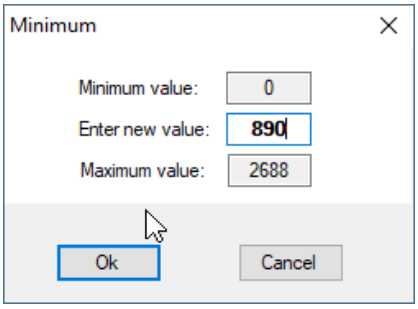


CursorX Centre Dead Zone value

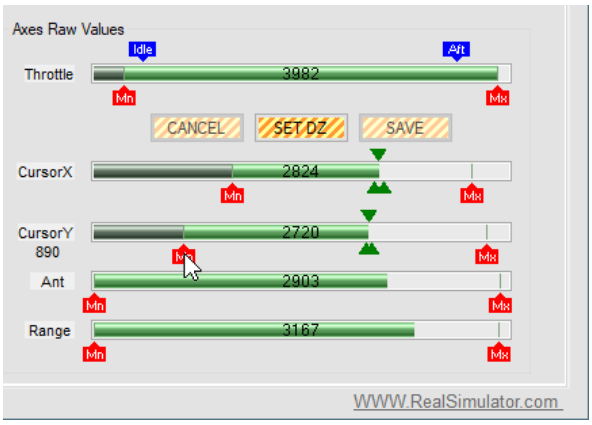
If you want to modify a value, with the mouse double-click on the icon and a new window will be opened showing the “minimum” (upper position) and “maximum” (lower position) admitted values for the selected parameter. In the centre position is showed in bold the actual value, clicking with the mouse in the box you can write the new value.



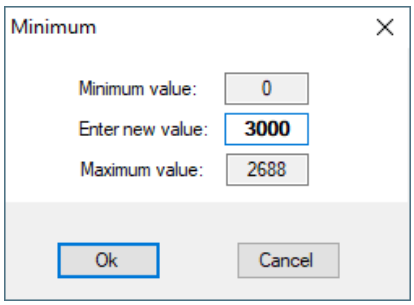
To finish the operation, click on the OK button to accept the written value.

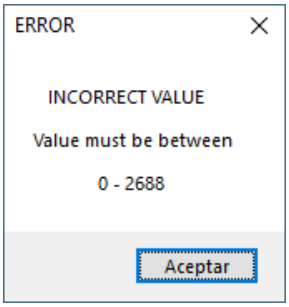


If the value is correct the window will close and the parameter and icon position in the progress bar will be modified.



If the value written is incorrect an ERROR window will be showed to inform about the error and the operation will be cancelled after accept the error.

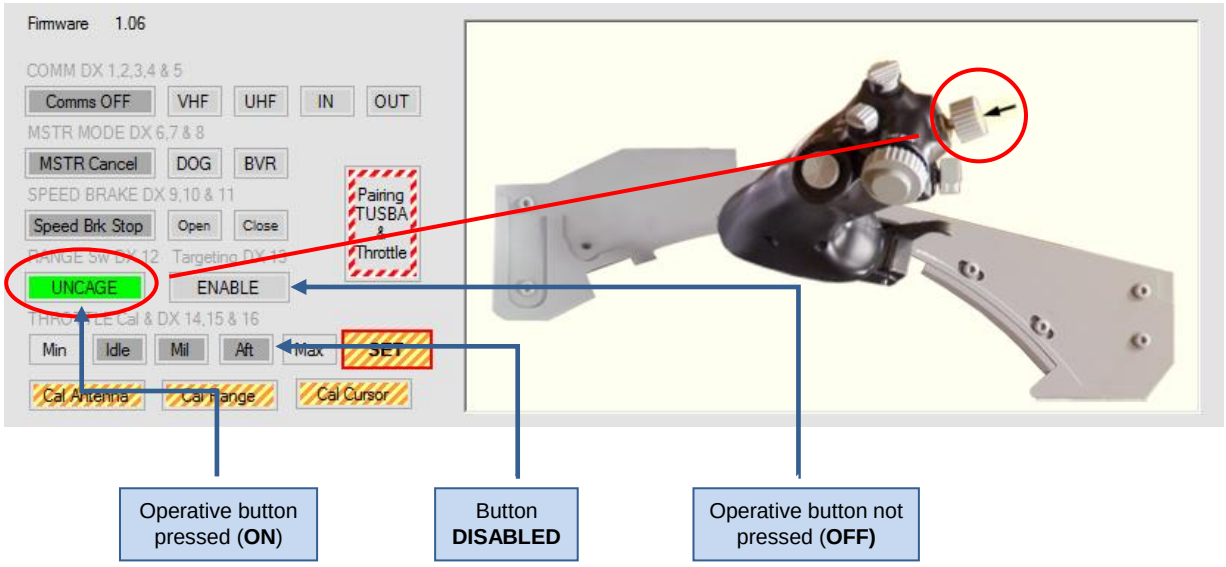




DX Buttons

Normally DX buttons show the status of buttons or switches when they are pressed or released, but TUSBA has been designed to show until 22 DX buttons more, all them can be enabled or disabled to avoid configuration problems in game setup.

As you can see below, every button is identified with its name and the DX number assigned; when the real switch or button is pressed it changes to green and it is showed with a black arrow in the animated area.



In general, colours of DX buttons are:

- Light grey: for operative buttons not pressed.
- Green: for operative buttons pressed.
- Dark grey: for buttons in disable state.

Next, you can see the 32 DX buttons of TUSBA description and how they work:

a) 10 standard DX buttons of Cougar. These buttons are:

DX 2 -> VHF
DX 3 -> UHF
DX 4 -> IFF In
DX 5 -> IFF Out

DX 7 -> DOG FIGHT
DX 8 -> MSL_OVRD

DX 10 -> Speed_Brake_OPEN
DX 11 -> Speed_Brake_CLOSE

DX 12 -> UNCAGE

DX 13 -> RDR CURSOR_ENABLE

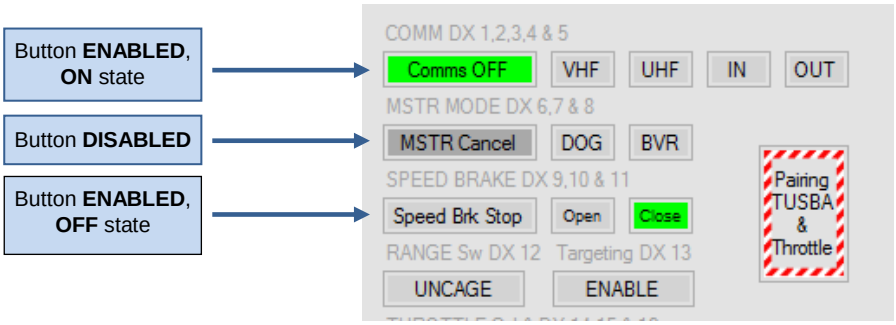
b) 3 additional DX buttons assigned to the central position of Comms, Dogfight and Speed Brakes switches. These buttons can be enabled and disabled with the left mouse button by clicking in the button. If the switch is released in the center position and the button is enabled, the button lights in green. These buttons are:

DX 1 -> Comms Off

DX 6 -> MSTR_Cancel

DX 9 -> Speed_Brake_Stop

In the next image you can see the different button's status:



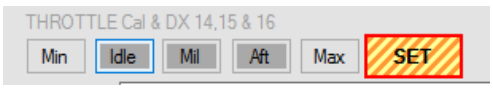
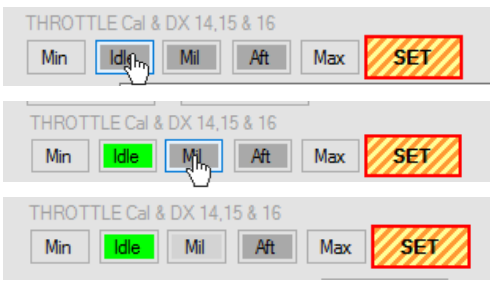
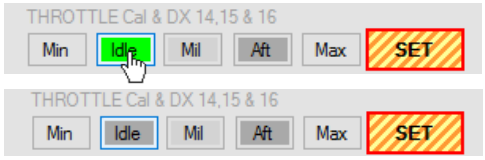
c) 3 additional DX buttons associated to the three throttle areas: Idle, Mil and Afterburner. These buttons are enabled and disables individually with the right mouse button by clicking in the correspondent button (remind the left mouse button is used in this buttons to assign the Idle and Afterburner positions) and alternatively they can be enabled and disabled at the same time by the SET/RESET button. These buttons are:

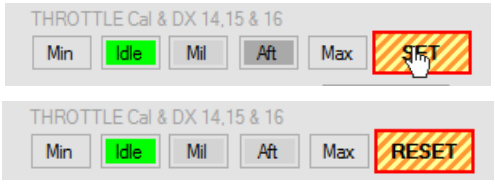
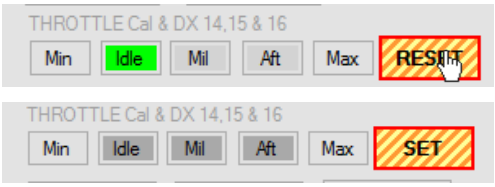
DX 14 -> Throttle Idle area

DX 15 -> Throttle Mil area

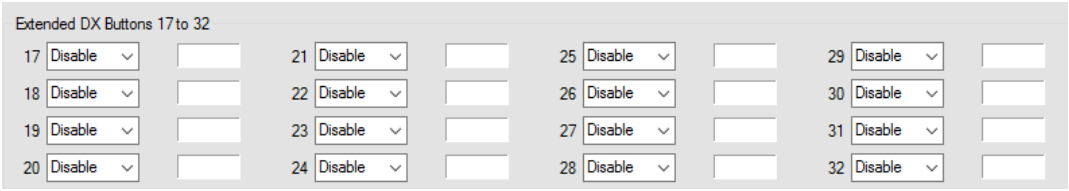
DX 16 -> Throttle Aft area

You can see here a sequence of how to configure them:

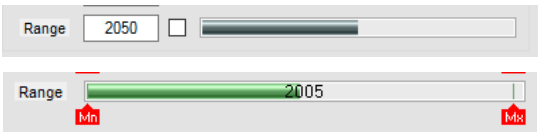
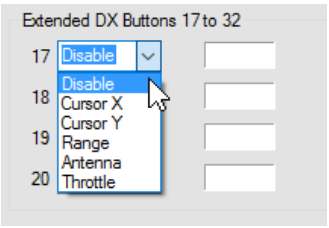
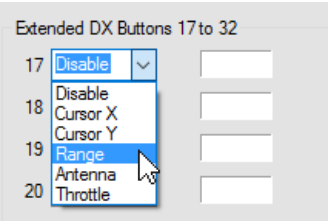
ENABLE/DISABLE INDIVIDUALLY BUTTONS	
We start with no DX button enabled (the three buttons are in dark grey)	
We are going to enable only the **Idle** and **Mil** DX buttons, so click with the right mouse button the Idle and Mil buttons. The Idle button is lighted in green because the throttle is in that position.	
Now, we are going to disable the **Idle** button, so click with the right mouse button the Idle button. Now Idle is in dark green.	

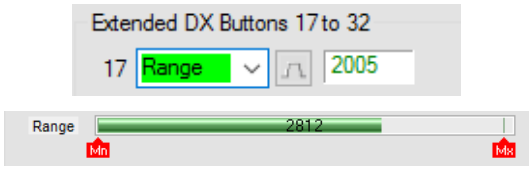
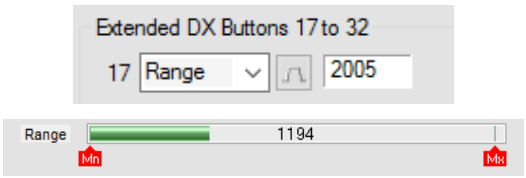
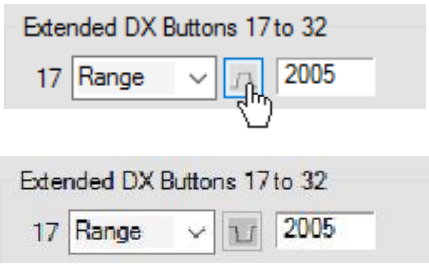
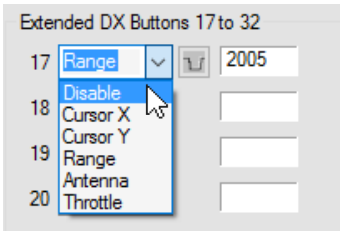
ENABLE/DISABLE JOINTLY BUTTONS	
If you want to enable the three buttons at the same time, click with the left mouse button the **SET** button to enable them. After pressing the button, it changes to **RESET**.	
If you need to disable the three buttons press the **RESET** button. After pressing the button, it changes to **SET**.	

d) 16 additional DX buttons linked to analog variables. These DX buttons are DX17 to DX32 and they are disabled by default; you need to configure them to be operatives.



Next, we will see an example of how to configure the DX17 button.

ENABLE/DISABLE DX BUTTONS LINKED TO ANALOG VARIABLES	
We are going to configure this DX button for active when the **Range** axis is greater than 50% (2048 in the DX axis measure and a raw value of 2005). So, move the **Range** rotary until this value aprox.	
Expand the combo box associated to the **DX 17 button**.	
Select the **Range** axis.	
And the button will be configured (for greater by default). As you can see the value for comparison is the raw value of 2005. Note that TUSBA does not work with the Windows calibration values, only with raw values.	

Now, if the Range is greater than 2005 the button status is ON.	
And if it is lower is OFF.	
If we wanted to invert the button status, that is to say, button active when is lower than the 50%, we would need to change the icon of **Active High/Low** to active low by clicking it.	
Finally, if we wanted to disable the button, we would need to expand the combo box and select the **Disable** option.	

PICTURE

DESCRIPTION

Changes from v1.07 (01/03/2016) to v1.07.1 Revision **Page**

PICTURE

DESCRIPTION

In this section, you will find answers to frequently asked questions. If your question is still unsolved, please feel free to contact us.

FAQ Links:

- I have just received my RealSimulator device and I want to install it, but I do not find the drivers in anyplace.
- I can’t update the firmware of my device with DCC.
- I want to install a new version of DCC or RS_HID_DEV_TOOL, but I cannot uninstall the older one.
- How to integrate TUSBA DX buttons in Falcon BMS?
- My cursor is in the central position but their associated DX measures are not centred.
- When I move an analog axis, I have spikes in the measure.
- How to know if when a switch or axis of throttle does not work the problem is in TUSBA or in the throttle hardware.
- My device has suddenly stopped working after connecting it or turning on the computer.

I have just received my RealSimulator device and I want to install it, but I do not find the drivers in anyplace.

Don't worry, no drivers are necessary, FUSBA uses standard HID drivers included in your installed operating system. No matter if x32 or x64, XP, W7 or other higher MS operating system. FUSBA will work in all situations.

RealSimulator provides two tools to configure and upgrade the device:

- Device Control Center (DCC): is the firmware update tool for the RealSimulator devices.
- RealSimulator HID Device Tool (RS_HID_DEV_TOOL): is a GUI application to configure and calibrate RealSimulator devices.

You can find more information about them in this User Guide in theirs correspondent paragraphs.

I can't update the firmware of my device with DCC.

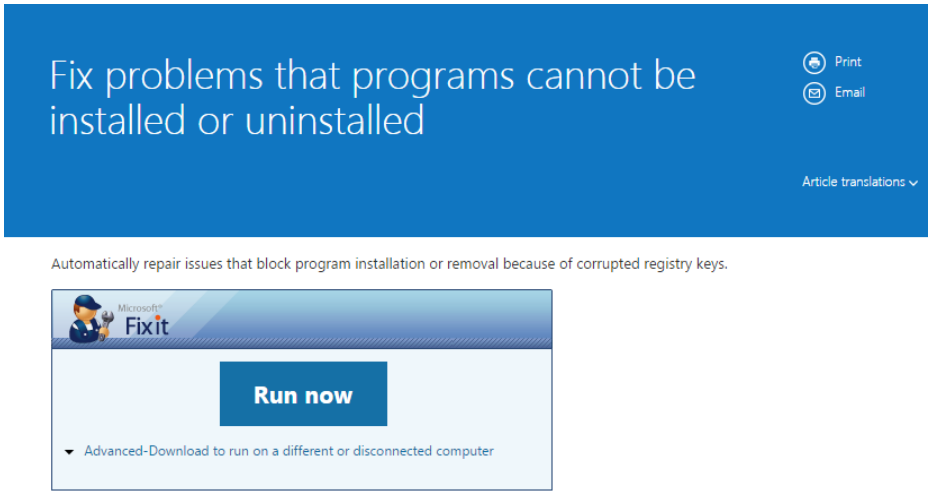
If you have followed the procedure given in the **Firmware Update** chapter and DCC cannot synchronize with your device to launch the update sequence, perhaps the source of problem is how you have connected the RS device to the computer. Please, connect the device with the supplied extension USB wire directly to an USB 2.0 port in the computer and try again.

I want to install a new version of DCC or RS_HID_DEV_TOOL, but I cannot uninstall the older one.

Microsoft has created a great and free tool to fix these problems. You can find it in the next link:

https://support.microsoft.com/en-us/mats/program_install_and_uninstall

Click the previous link (or next picture) to open the webpage and press the **RUN NOW** button to launch the wizard.



How to integrate TUSBA DX buttons in Falcon BMS?

Dunc has written a document where you can find all the necessary information to integrate TUSBA buttons, and any DX device, in Falcon BMS. You will find this document in your BMS installation, located in the next path in function of the BMS version installed in your computer:

- BMS 4.32: C:\Falcon BMS 4.32\User\Joystick\Cougar\Dunc_DX
- BMS 4.33: C:\Falcon BMS 4.33\Docs\Key Files & Input\Dunc_DX

To facilitate the work, we give you an example code ready to “cut & paste” in your .key file, although it is possible you need to modify it as you will see below.

```
#================================================================
#=============  RealSimulator  TUSBA  -  DX buttons  ==============
#================================================================
SimDoNothing 0 -1 -2 0 0x0 -1 "COMMS Switch OFF"
SimTransmitCom2 1 -1 -2 0 0x0 -1 "COMMS Switch Down - VHF"
SimTransmitCom1 2 -1 -2 0 0x0 -1 "COMMS Switch Up - UHF"
SimCommsSwitchRight 3 -1 -2 0 0x0 -1 "COMMS Switch Right - IFF IN"
SimCommsSwitchLeft 4 -1 -2 0 0x0 -1 "COMMS Switch Left - IFF OUT"
SimDeselectOverride 5 -1 -2 0 0x0 -1 "DOGFIGHT Switch - MRM/DF Cancel"
SimSelectSRMOverride 6 -1 -2 0 0x0 -1 "DOGFIGHT Switch - DF Override"
SimSelectMRMOverride 7 -1 -2 0 0x0 -1 "DOGFIGHT Switch - MRM Override"
AFBrakesToggle 8 -1 -2 0 0x0 -1 "SPD BREAK Switch - Toggle"
AFBrakesOut 9 -1 -2 0 0x0 -1 "SPD BREAK Switch - Open"
AFBrakesIn 10 -1 -2 0 0x0 -1 "SPD BREAK Switch - Close"
SimToggleMissileCage 11 -1 -2 0 0x0 -1 "MAN RANGE Knob - UNCAGE"
SimCursorEnable 12 -1 -2 0 0x0 -1 "RDR CURSOR - Cursor Enable"
#================================================================
```

To understand this code correctly you must take into account that:

- Windows count the buttons from 1 to 32.
- BMS count the buttons from 0 to 31 in the .key file.

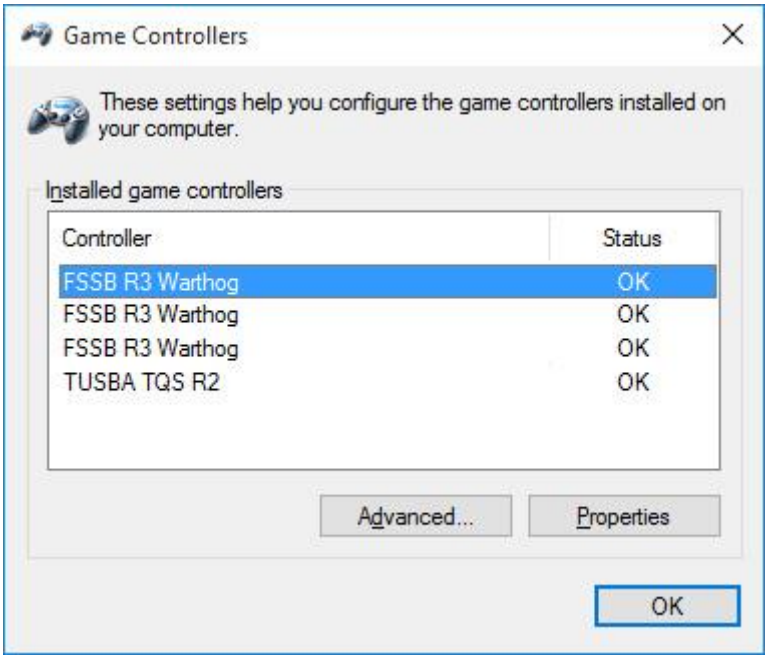
So, we can see that the VHF button (for example) is identified in windows as DX 2 button of TUSBA and in the previous code as number “1”.

```
SimTransmitCom2 1 -1 -2 0 0x0 -1 "COMMS Switch Down - VHF"
```

You also need to know that in function of TUSBA device position has in the Game

Controllers window you will need to renumber the button numbers in the previous code, so those numbers are only valid if TUSBA is the first device, but if it is the second you will need to add “32” to the numbers of above code, if it is the third you will need to add 64, and so on, you will need to add 32 for each previous device in the list.

To see TUSBA position in the device list, plug in all the devices that you use to fly and open the Game Controllers window.



As you can see, in this example TUSBA is the 4 device, so you will need to add to each number “96”, and the definitive code to paste will be:

```
#================================================================
#============  RealSimulator  TUSBA  -  DX buttons  ==============
#================================================================
SimDoNothing 96 -1 -2 0 0x0 -1 "COMMS Switch OFF"
SimTransmitCom2 97 -1 -2 0 0x0 -1 "COMMS Switch Down - VHF"
SimTransmitCom1 98 -1 -2 0 0x0 -1 "COMMS Switch Up - UHF"
SimCommsSwitchRight 99 -1 -2 0 0x0 -1 "COMMS Switch Right - IFF IN"
SimCommsSwitchLeft 100 -1 -2 0 0x0 -1 "COMMS Switch Left - IFF OUT"
SimDeselectOverride 101 -1 -2 0 0x0 -1 "DOGFIGHT Switch - MRM/DF Cancel"
SimSelectSRMOverride 102 -1 -2 0 0x0 -1 "DOGFIGHT Switch - DF Override"
SimSelectMRMOverride 103 -1 -2 0 0x0 -1 "DOGFIGHT Switch - MRM Override"
AFBrakesToggle 104 -1 -2 0 0x0 -1 "SPD BREAK Switch - Toggle"
AFBrakesOut 105 -1 -2 0 0x0 -1 "SPD BREAK Switch - Open"
AFBrakesIn 106 -1 -2 0 0x0 -1 "SPD BREAK Switch - Close"
SimToggleMissileCage 107 -1 -2 0 0x0 -1 "MAN RANGE Knob - UNCAGE"
SimCursorEnable 108 -1 -2 0 0x0 -1 "RDR CURSOR - Cursor Enable"
#================================================================
```

My cursor is in the central position but their associated DX measures are not centred.

As we talk in this user guide, to obtain the best performances of your device is very important to make the pairing process and axes calibration correctly. We describe step by step how to do it in the **TUSBA Set up** charter.

If you have followed this procedure and you continue having problems with the DX measures of cursor because they are not centred when the cursor is released and

centred, possibly the source of this problem is the mechanic play of plastic parts and the potentiometers accuracy and repeatability. You can solve it modifying manually the dead zone of axis with problems. You can see how to do it in the **Calibration of an axis with central position** section of **TUSBA Set up** charter.

When I move an analog axis, I have spikes in the measure.

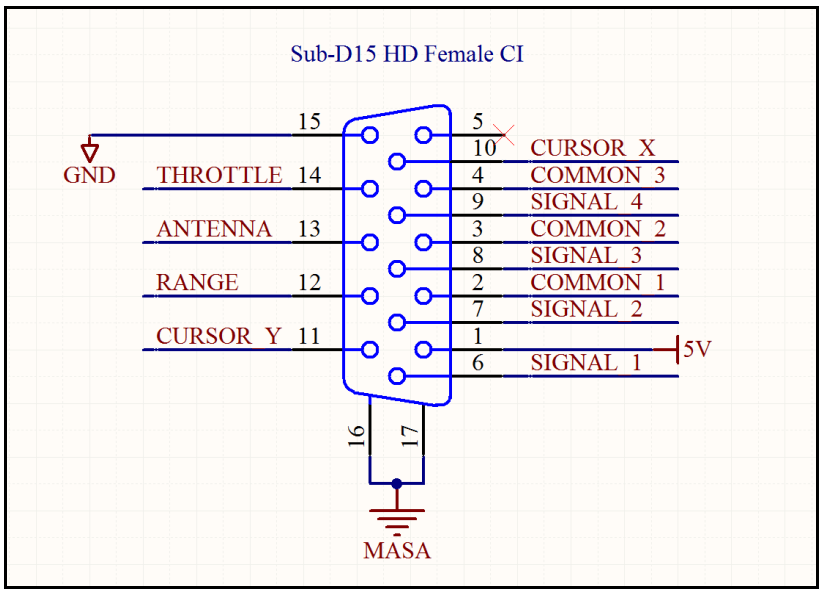
Analog measures in all RealSimulator devices are digitally filtered, so if you have spikes in measures of one analog axis of your device, the problem is the potentiometer and you will need to change it, so you must look for a spare part.

How to know if when a switch or axis of throttle does not work the problem is in TUSBA or in the throttle hardware.

Always that a switch do not work when is pressed or an analog axis when is moved, we have a doubt about where the problem is, on the controller board (TUSBA) or in the hardware (wires, sensors, buttons and switches) of Cougar Throttle.

If you still have the Cougar stick a good and quick option is connect the throttle to the stick and test if the switch or axis works properly, but if you do not have the stick we give you following an easy procedure to found the damaged part.

You will only need a small wire to manually connect pins in the TUSBA Sub-D connector. The next image shows you the pin-out of this connector.



To start the test, unplug the throttle from TUSBA and plug in TUSBA in an USB port of computer through the supplied extension cable. Finally launch the RS_HID_DEV_TOOL program and follow the next instructions:

a) If the problem is an axis, identify in the previous image the pin number of axis to test and first, with the wire join that pin to the “GND” signal (pin 15) and verify in the program window if the measure bar associated to the axis goes to minimum, and second, join the pin of axis to the “5V” signal (pin1) and verify if the axis goes to maximum. If this happens just like that, all is well in TUSBA and the problem is in the throttle hardware, if not the problem is in TUSBA and you need to send it to

RealSimulator technical service to repair.

For example, if the problem is the **Range** axis, you will join first the pin 12 and 15, and in second place, the pin 12 and 1.

b) If the problem is a switch that does not work, locate in the next table the name of switch and get the associated pins in the row and column header to the switch, then join with the wire the two pins and if the button in the RS_HID_DEV_TOOL lights in green, TUSBA is well and the problem is in the hardware, if not the problem is in TUSBA and you need to send it to RealSimulator technical service to repair.

		COMMON		
		1 (pin 2)	2 (pin 3)	3 (pin 4)
SIGNAL	1 (pin 6)	VHF	DOGFIRE	UNCAGE
	2 (pin 7)	UHF	BVR	ENABLE
	3 (pin 8)	IN	OPEN	----------
	4 (pin 9)	OUT	CLOSE	----------

For example, if the **Speed Brake Open** switch does not work, you will need to join the pin 8 and 3.

We hope you have no problems with your hardware, but if you have it, this easy procedure will be able to simplify the location of problem.

My device has suddenly stopped working after connecting it or turning on the computer.

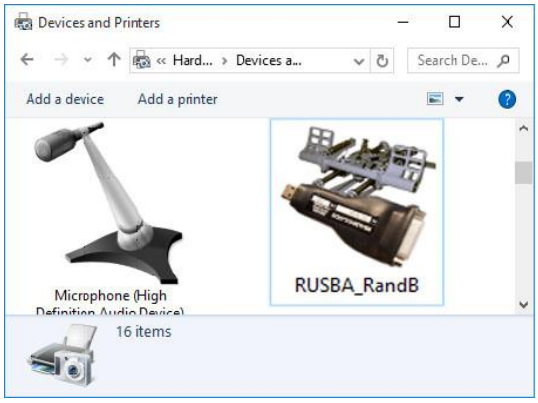
As the title says, if sometime your device does not work after connect it to the computer or after a power on and you are running Windows 10, then please, read this paragraph.

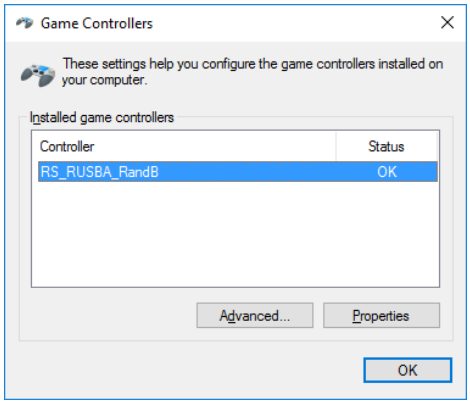
Windows 10 has a bug with the HID composite devices, sometimes it changes the HID devices order and when programs access to the device information selected, it is not correct.

For these occasions we suggest follow this method, we have used it when the problem has occurred and usually the problem is solved.

Since the problem occurs very rarely and we have not got TUSBA screenshots, the next explanation is done with RUSBA screenshots.

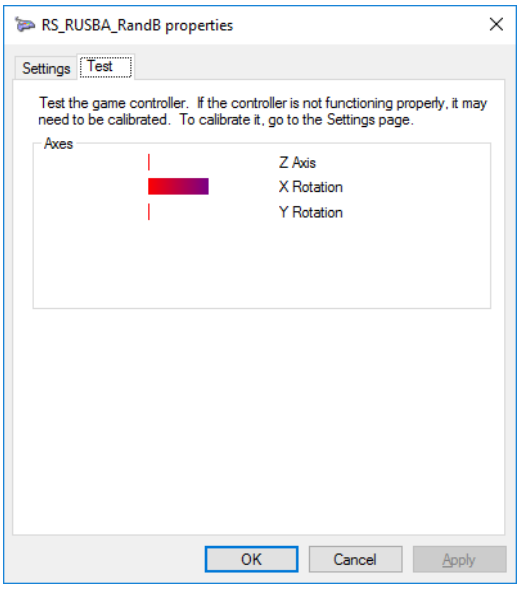
Open the **Devices and Printers** window, click with the right mouse button the **RUSBA_RandB** icon and select the **Game controller settings** option in the pop-up menu to open the **Game Controllers** window.

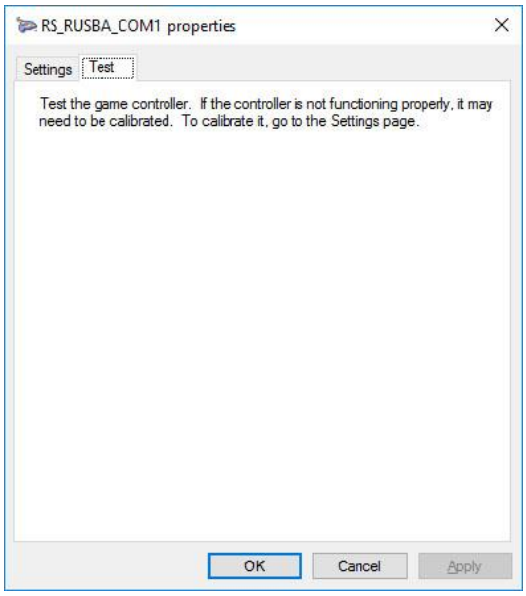




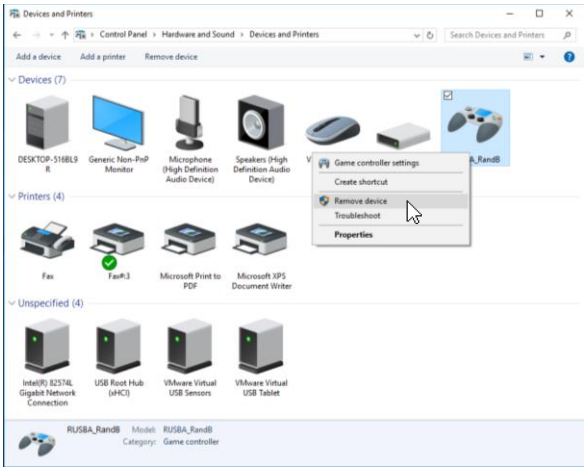
Click the **RS_RUSBA_RandB** text controller once to highlight it as shown in the previous image and next, click on the **Properties** button to open the **RS_RUSBA_RandB** properties window; you must have a window as this where you can see the four axes and the four buttons status.

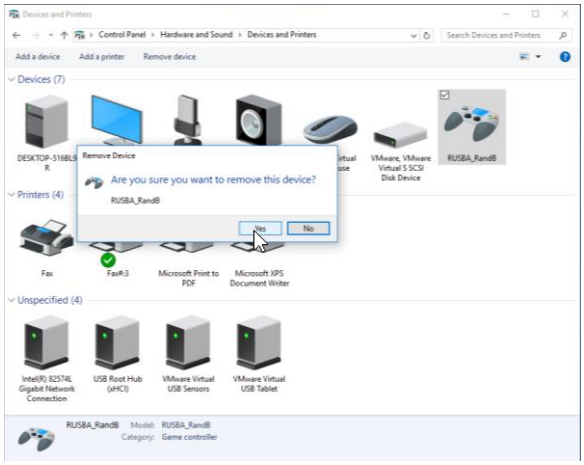
You should have a window as next left image with the axes bar graph, but if the system is wrong, you will see an image as right one. If you look at the imagen you will check the error, the window name is not correct and the axes area does not exit.





To solve the problem you must remove the device, for that, go to the the **Devices and Printers** window and click again with the right mouse button over the **RS_RUSBA_RandB** icon and select the **Remove Device** option. Finally, press the **Yes** button in the next confirmation window.





When the device is removed, unplug the device and after some seconds plug again the device and verify if the problem is solved. Usually the problem will be fixed, if not, repeat the procedure.