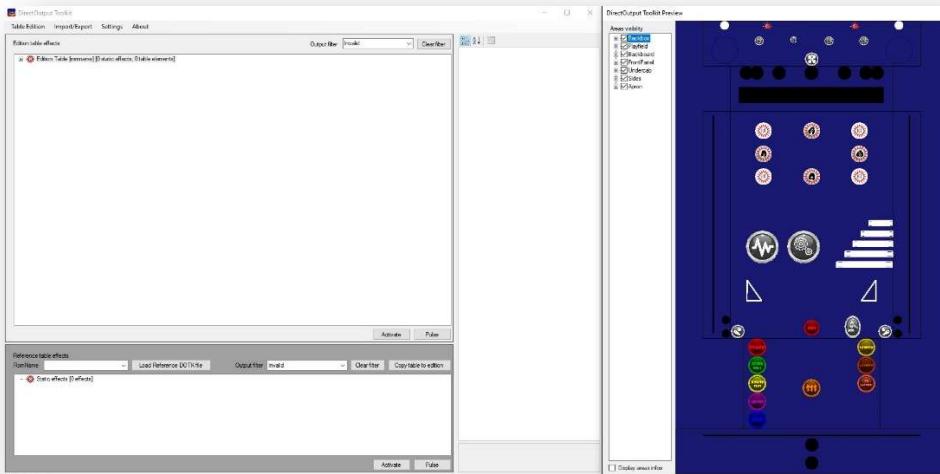


# Direct Output Toolkit



## What is DOTK ?

Direct Output Toolkit (DOTK) is a tool for editing the Dof setup of a table without having to go back and forth with [DofConfigTool](#) and your VPinball.

DOTK is based on the DirectOutput framework (<https://directoutput.github.io/DirectOutput/>), so it handles effects exactly the same way the Dof would.

Its advantage is that it is completely standalone, it contains its own fully configurable preview window which will allow you to immediately see the result of the effects you are editing.

There are some videos presenting the different possibilities of DOTK on my channel (<https://www.youtube.com/playlist?list=PLjHwWo3oP7bjyNVtPrCltQ-fieKkx6iYk>)

## Installation & first setup

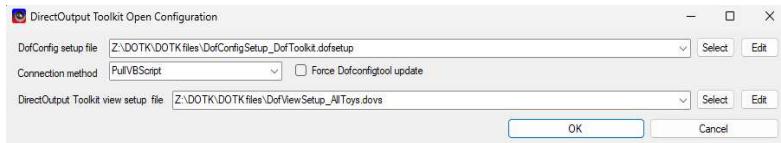
You will find the DOTK directly on my git releases page  
<https://github.com/Vroonsh/DirectOutput/releases>

DOTK installation is very easy, you just have to extract the contents of the DOTK.zip file into a directory and run DirectOutputToolkit.exe.

You don't need to be in the DirectOutput directory to work, the setup files and directoutputconfig.ini will also be fetched locally, no need to use yours.

DOTK will automatically generate a local globalconfig and it doesn't need cabinet.xml to handle addressable effects either.

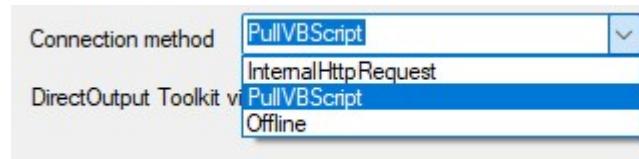
The first time you run DOTK you will have this setup window.



Here you will be able to choose your setup files to run DOTK.

**DofConfig setup:** this file will allow you to declare the setup that you used for your DofConfigTool config (including the APIKey), you can create one that corresponds to your DofConfigTool account or use the one provided for the DofToolkit account which is already ready for use with all available outputs.

**Connection Method :** You can choose which method you'll use to retrieve your dofconfigtool ini files from the site.



You can choose between these options :

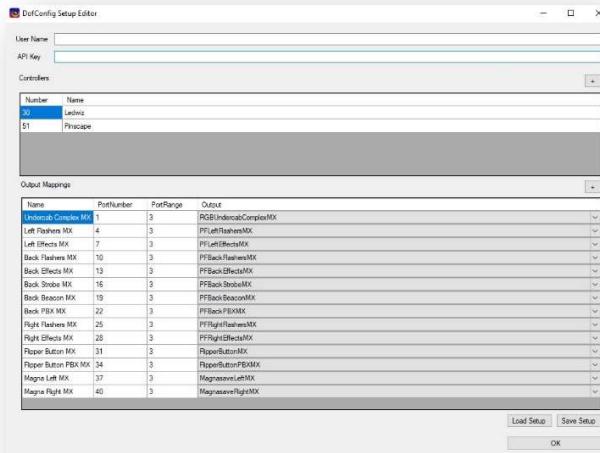
- **InternalHttpRequest** : former method, not always able to connect to the site
- **PullVBScript** : use the vbs script for dofconfigtool site, you'll have to update it if it changes it's in DOTKFiles/VB, this method always force the update.
- **Offline** : Doesn't do the update

**Force DofConfigTool update :** by checking this box you will force the download of the DofConfigTool config linked to the setup you have chosen; this is useful if there has not been an update but you have changed the outputs.

**DirectOutput Toolkit view setup:** this file will describe the different toys you want to have in your preview window. Here too, you can create your own or use one of the provided ones (Full toys or addressable only).

It is also in this window that you will be able to edit your own setup files by clicking on the **Edit** buttons.

## DofConfig setup edition



To edit your own setup, use this editor.

You just have to provide a username name (in fact you can put whatever you want it will not be used, it's just to ease if you have severals) and an API Key which will be used to retrieve the .ini files locally so that DOTK can show you the effects of reference tables.

Then you can redo the setup you have on DofConfigTool so that DOTK can recover with your .ini files. A setup already ready for the DofToolkit user is already provided, it covers all the outputs on two controllers (a pinscape 51 and a ledwiz 30), I recommend this one if you want to create or upgrade a complete Dof intended to be published.

To do your setup, add a controller by giving it a name and the number of the corresponding .ini file.

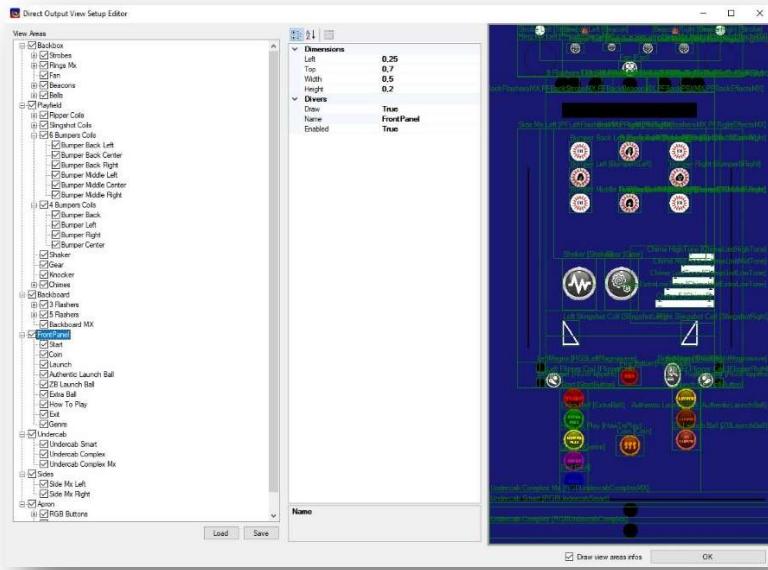
Device: WS2811 1 - directoutputconfig 30 ▾

For each of these controllers, you can declare outputs exactly as in the DofConfigTool window. Port number and type of output (the range will be calculated automatically).

You can delete either an output or a whole controller by selecting it and press the **Delete** key.

If you have correctly matched your DofConfigTool outputs with those of this setup, the .ini files retrieved for your account will work as if they were on your cab.

## DirectOutput Toolkit view setup edition



It is with this editor that you can create a setup for your preview window that best matches your installation.

Two setups are already provided which cover almost all toys or just addressable toys if you just want to focus on the Mx.

You can start from an existing setup and modify or create it from the beginning.

By right clicking in the treeview window, you can add Areas of three types. The Virtual, Analog and RGB areas.

Virtual areas are just containers in which you can put other areas, they have a name, dimensions (Top, Left, Width, Height) and indicates whether we will draw a frame for this area in the preview.

As for the coordinates of the areas, they are always in percentage of the parent area (this allows you to quickly resize an entire virtual area with everything in it).

Only virtual areas can contain other areas (including other virtual).

These are the other two types of areas that will really represent your toys. In addition to the name and dimensions they will also contain a list of outputs that they will react to when the DOF triggers them.

These areas will sometimes be represented by icons which correspond to the first output of the indicated list.

Area Analogs are those that correspond to analog toys (1 output) such as front panel buttons, bumpers, slingshot, shaker ...

You will be able to say whether their representation is squared or not and possibly choose a background color (practical to differentiate the different front buttons for example).

The RGB areas will represent all the toys that use color (3 outputs), this includes flashers, undercabs, Mx toys...

To configure an RGB toy is like analog except that you will indicate what type of toy it will be (ValueType either Single or Addressable) and how it will be rendered (Simple, Matrix, Ring, Frame).

In the case of Matrix and Frame it will be necessary to say the width and height in leds, for the Ring just the length and the angle from which the ring starts (0 is on the right then in the counter-clockwise direction of watch).

This editing process is quite long, if you don't need a custom config, I advise you to use the two setups already provided.

### DOTK preferred setup regarding usage

First, what you should know is that I have split the dofsetup & dof view setup so you can have any combination of specific visualization of the same dofsetup. Dofsetup & dovs files are totally independent one from each other.

That's why I provided several dovs files with only one dofsetup file.

If you plan to use DOTK to create or modify dof tables which will have to be uploaded to DofConfigTool, you should consider using the DofToolkit user dofsetup file because it covers all available inputs from DofConfigTool without any combos or duplicates. Using this setup will ensure you the exported dof lines will really be designed for a particular output and so you'll be able to confidently paste it into the corresponding line in your Table settings on the site.

I made DOTK firstly to reduce iteration time while creating your effects, its purpose is not to replace DofConfigTool about config files generation (the site already does it perfectly and the community aspect is the most important).

So, using DOTK with the DofToolkit user setup then copy paste your modified lines to the site and finally generating your config files will allow you to test your modification on your vPinball.

In case you already have local modifications on your account for some tables you want to have in DOTK using the DofToolkit setup, you'll can easily import them using the import dialog described below.

One of the reasons you'll have to create your own dofsetup & dovs files would be, imo, to simulate a future real setup or validate your current setup in case of an upgrade or something.

### DofConfigTool data update

When you have finished choosing your setups and launch DOTK, you will have a window that will tell you that either you do not yet have any .ini files that correspond to the setup you have chosen, or that they aren't up to date. In the first one you will have to retrieve them at least once otherwise DOTK will not work, the update is not mandatory and you will be asked again at each launch. You will then have a window telling you that DOTK is retrieving the .ini files from DofConfigTool. These files will end up in a **setups\Username-APIKey** directory where your dofsetup file is located.

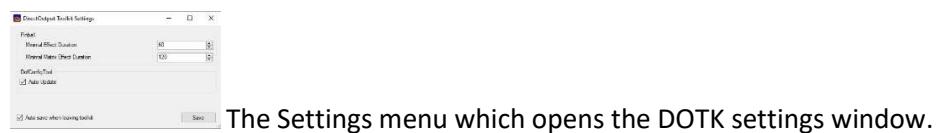
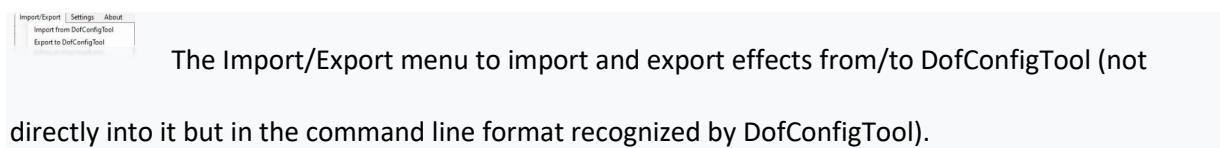
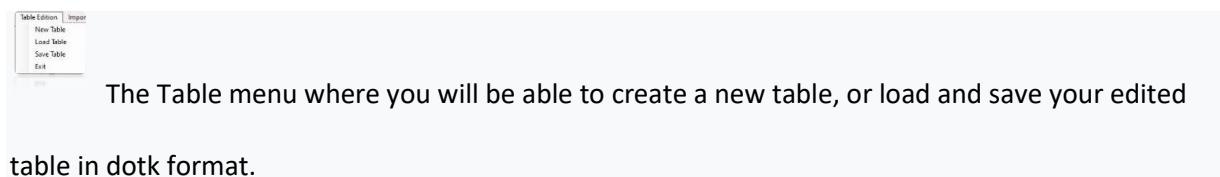


## Using DOTK

Once DOTK is launched, you will have two windows, the main window and the preview window.

They are separate so you can do whatever you want with size and placement (it's saved and reloaded in settings).

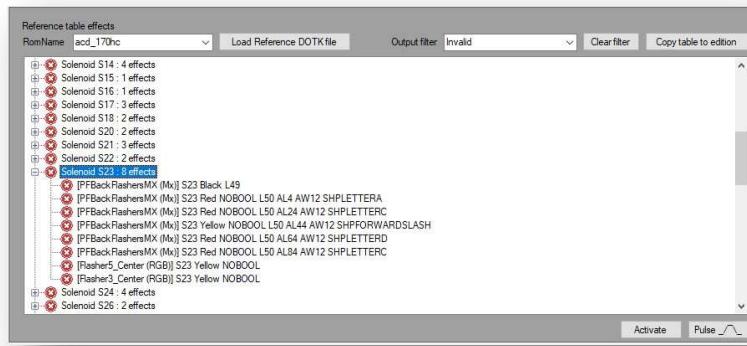
### DOTK Menus



### DOTK Main window

The main window of DOTK contains three distinct areas: The reference area, the editing area and the properties area.

## Reference area



The reference area will be where you can look at everything that has already been done from .ini files, you can also open other effect library dotk files.

This is where you can gather effects and table elements to fill your edit area.

The **RomName** combobox allows you to see the effects of the different tables already available on DofConfigTool.

The **Load Reference DOTK** button allows you to reload any existing dotk file for reference, for example this could be used to have dotk files of effect libraries ready to use.

Each time you select a TableElement or an effect, its properties will be displayed in the properties area but in read-only mode.

You can view a TableElement or an effect by pressing the **Activate/Deactivate** button or the **Pulse** button and see the effect in the preview window.

The pulse button plays the effect as long as the button is pressed, so you can pulse with quick clicks. The activate button will allow you to have multiple effects active at the same time.

When an effect is activated, the red icon next to the effect (or table element) turns green.

You can filter the treeview by output with the **Output Filter** combobox, you will only see the effects linked to the selected output.

If you activate a TableElement with an active filter and it contained other effects on other outputs, they will be played too even if not shown.

You have a **Copy table to edition** button which will copy all the contents of the current reference table to the edition table (by deleting what is already in the edition table).

This is a good starting point for upgrading an existing table.

By right clicking on the TableElement or the effects of the reference table you will have a context menu allowing you to make copies to the editing table.

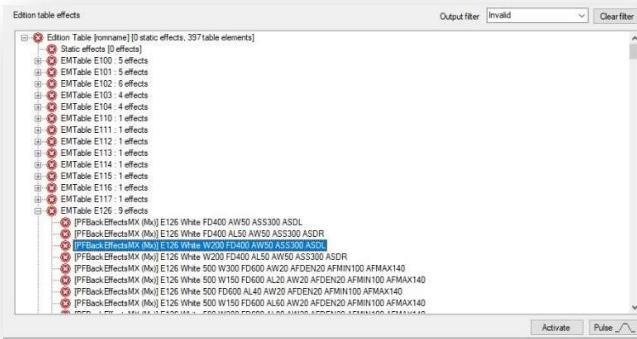


You can also do that by drag & dropping an effect or a table element from the reference window to the targeting node in the edition window.

For Effect nodes you also have a context menu to copy the Dof command of the effect to the clipboard.



### Edition area



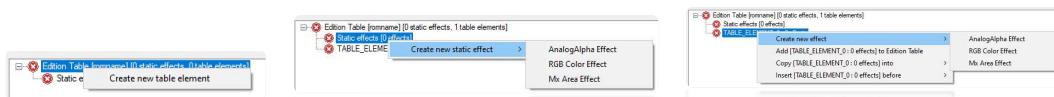
The editing area looks a lot like the reference area, you will find the **Output Filter** and the **Activate / Deactivate** and **Pulse** buttons there which will have the same effect.

In this area, if you select TableElement or effects, this time the properties will be editable in the properties area.

You will also be able to make copies of Effects and TableElement with the context menu, but this time within the edit table.

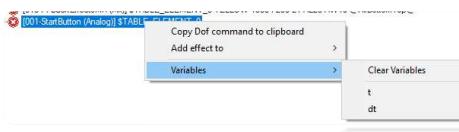
You'll also be able to drag & drop effects and table element within the edition window, by pressing the **Shift** key you'll move the node instead of copying it.

You can also create TableElements and effects using these context menu.



You can delete TableElement or Effects by selecting them and pressing the **Delete** key.

You can also assign variables to your effects, when an effect have compatible variables, they're displayed in the « **Variables** » contextual menu.



You can also edit the properties of your table, this will be done in the properties area.

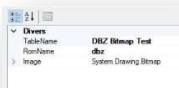
### Properties area



The properties area will allow you to change the parameters of your table, TableElement and different types of effects.

It updates each time you select something in one of the two reference or edit areas.

### Table Properties



**TableName** : the user friendly name of the table

**RomName** : the associated rom name (the one which will be used in the table settings .ini files)

**Image** : A specific image that will be used by this table for the MxBitmapEffects, when you choose an image in this property, it is automatically updated in the directory where the .ini are located to be usable by DirectOutput. It will use the name of the rom (entered in RomName) to create a romname.\* <extension> file. If you change the extension (for example you had a PNG and you choose a GIF), all the romname.\* images will be removed from the directory where the .ini are found to have only the one you have chosen (DirectOutput supports one image per table).

### TableElement properties



When you select a TableElement, you will have several types of properties depending on the type of element.

If **TableElementType** is **NamedElement**, you will have a Name field allowing you to indicate the name of the TableElement.

NamedElement are very rarely used.

For all other types, you will indicate a number.

### Effect properties

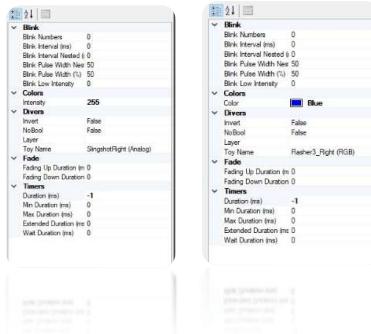
When you select an effect, depending on its type, you will have differences in properties.

All the effects will have properties in common:

- Miscellaneous section: Invert, NoBool, Layer and Assigned Toy
- Blink section: everything related to blinking
- Fade section: fade in / out effects
- Timers section: MinDuration, MaxDuration, Duration ...

The **ToyName** combobox will provide you with the list of toys compatible with your type of effects (Analog, RGB, Addressable).

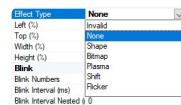
The analog or RGB effects will have almost the same properties with a difference, in analog one manages an intensity, in RGB a color.



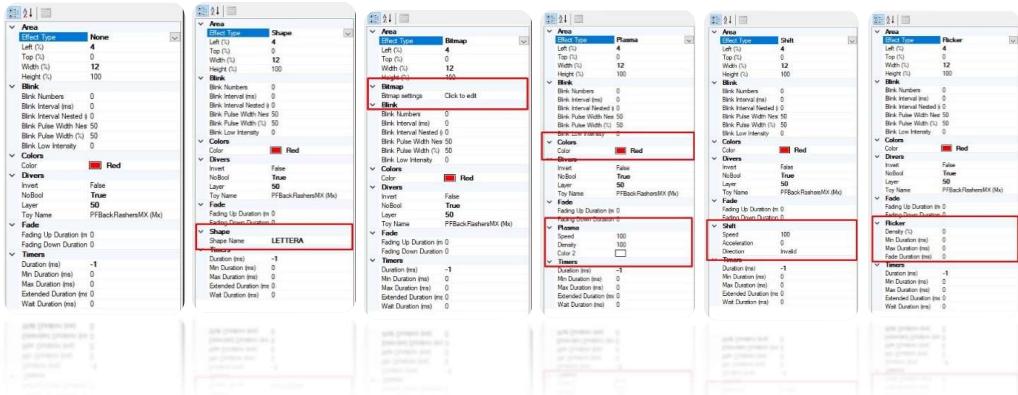
The intensity is in full range (0-255), it can be brought back to the classic range (0-48) at export time.

The basic colors (from the .ini) are available in the **Color** combobox but you can also indicate colors not provided for in the list by choosing Custom Color.

The Mx effects are a little more complex, you will first have to define an Mx effect type among those available:

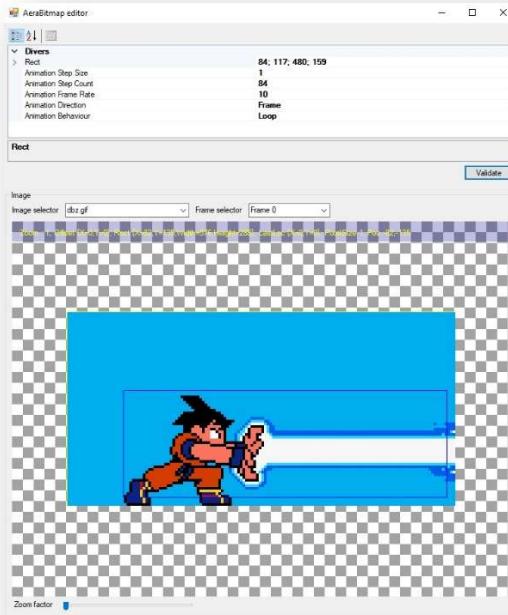


Then, depending on the type of Mx effect you have chosen, you will have a specific section by type of effect.



Small specific case for Mx Bitmap effects, to simplify the way to edit them, a special editor has been added.

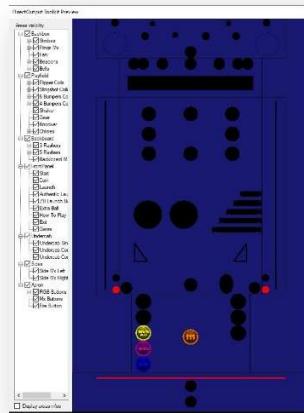
Clicking on **Bitmap Settings** in the properties will open this editor.



In this editor, you will have a real-time view of the part of the bitmap that you have selected, including whether you have specified any animation parameters.

Once validated, you will be able to see the result directly on your Addressable Toy by activating the effect.

## Preview window



The preview window is where you will see your effects in real time when you activate effects or TableElement.

Contains two main areas: the actual preview area and the visibility area.

In the visibility area you will be able to show or hide entire parts of the preview, this allows you to better concentrate on what you want to edit.

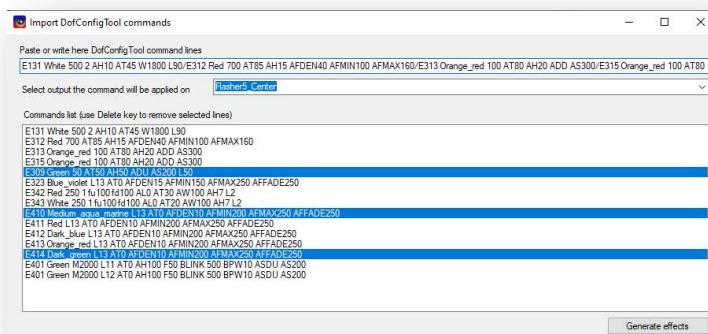
The **Display Area Info** checkmark will show you the data that is usually shown in the editing phase of this setup.

By passing the mouse over an area in the preview you will have the list of outputs assigned to it.

## DofConfigTool Import/Export

You have the option of importing or exporting Dof command lines (the famous lines that not many people understand 😊) into DOTK.

For that you have two dedicated windows.



The import window will allow you to copy a line from DofConfigTool like these for example.

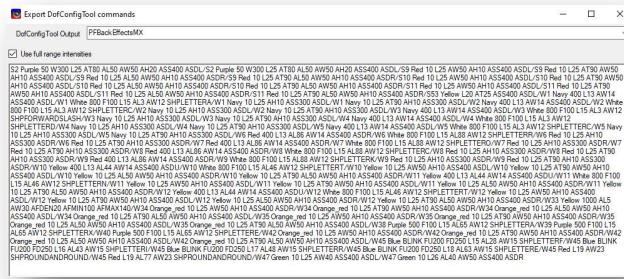
Kaleidoscope Complex MX	E101 Blue AT0 AL0 AT0	Kaleidoscope Complex MX	E111 Prof AL0 AT0
PF Left Flashes MX	S10 Blue AT0 AH12 L0\$19 Green AT0 AH12 L0\$26 Red f100 AT0 AH12 L	PF Left Flashes MX	S10 Blue AT0 AH12 L0\$18 Green AT0 AH12 L1\$26 Red f100 AT0 AH12 L
PF Left Effects MX		PF Left Effects MX	
PF Back Flashes MX		PF Back Flashes MX	
PF Back Effects MX	jsn@ L2\$29 White f100 @fshemuro@ L3W08 Red f20 @fshemuro@ L4	PF Back Effects MX	S16 Blue @tshemuro@ L1\$18 Green @tshemuro@ L2\$28 Red f100 @f
PF Back Stroke MX	E100 AT0 AH12 A W0100 AS0L ASS200 BLNK 490 BPW15 1601-19	PF Back Stroke MX	S11 Blue AT0 AH12 ALW AY0100 AS0R AS220 BLNK 490 BPW15 F100 L
PF Back Stroke MX	S1 @shn@ 300 L1\$18 @gshn@ 300 L1\$25 @eshn@ 300 L1\$56 @e	PF Back Stroke MX	S3 @shn@ 300 L1\$14 S @shn@ 300 L1\$25 S @shn@ 300 L1\$56 S @e
PF Back Stroke MX		PF Back Stroke MX	
PF Back PMS MX	Blue ALD AW18 FU1200 FD150 SHPLTWR13 Blue ALD A20 AW19 FU1200 FC	PF Back PMS MX	Blue ALD AW18 FU1200 FD150 SHPLTWR13 Blue ALD A20 AW19 FU1200 FE
PF Right Flashes MX	S19 Red AT0 AH12 L0\$27 Red f100 AT0 AH12 L0\$28 White f100 AT0 AH	PF Right Flashes MX	S19 Red AT0 AH12 L0\$27 Red f100 AT0 AH12 L2\$28 White f100 AT0 AH
PF Right Effects MX	W16 Blue AT0 AH15 AF0EN20 AFMIN100 AFMAX160 W16 Red AT05 AI	PF Right Effects MX	W16 Blue AT05 AH15 AF0EN20 AFMIN100 AFMAX160 W16 Red AT05 AI
Fliper Button MX	ON Blue AT0 AL0 AW100 AH100	Fliper Button MX	ON Blue AT0 AL0 AW100 AH100
Fliper Button PBX MX		Fliper Button PBX MX	

You can then paste them into the designated field in the import window and you will immediately see the different effects listed below.

You can then keep only some of them by deleting the ones that do not interest you.

Finally, you just have to choose which output you want to import and validate and the effects and TableElement will be automatically created.

For the export part it's the same thing, you have a dedicated window.



This window will take all the effects that are in your edit table and transform them back into DofConfigTool command lines classified by output. You just have to choose the output that interests you and copy the content of the textbox to copy it to the row you want in the DofConfigTool.

The **Use full range intensities** check box allows you to return to the range 0-48 for analog intensities if you wish when it is unchecked.

The Dof should normally support full range intensities (0-255) which gives you a bit more finesse but it's less standard.