# Programming the Multiplex Profi mc3030

by Mike Shellim

*Sponsored by*
**Multiplex USA**
560 Library Street
San Fernando, California 91340
Phone (818) 838-6467
Fax (818) 838-3127
http://www.multiplexrc.com

Version: 2.2

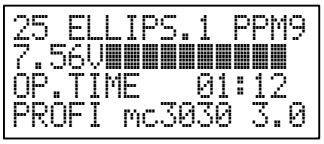
Please read licence conditions on page 2

# Contents

# Introduction to 2nd Revision

Welcome to the "alternative" guide to programming the Multiplex mc3030. The purpose of the guide is to enable you to configure your radio from scratch, without recourse to the user manual. If you're a 4000 user, you may also benefit from the guide since the basic architecture is similar to the 3030. If you don't have a Multiplex radio yet, the guide will give you a taste of the unique features and flexibility of the system.

A central feature of this guide is a three-step procedure which allows even complex set-ups to be programmed with the minimum possibility of error.




Hopefully you will find the style more concise than the user manual. Since the emphasis is on programming, I've omitted any reference to housekeeping topics such as airborne equipment and NiCd maintenance – these are dealt with adequately in the user manual. Neither have I covered helicopter-specific features, however the same principles apply to all kinds of model.

A basic knowledge of computer radios is assumed. If you're a complete beginner to computer radios you may benefit from running through the mc3030 user manual first, then come back here to do some *real* programming!

This 2nd revision has an expanded section on user mixers. I'm fairly sure there will be no more major revisions!

Incidentally, if you are not yet an owner of a 3030, there is a detailed review available on the R/C Soaring web site at http://www.rc-soar.com along with users feedback and tips.

## Licencing

A very quick word about licencing. This guide for free for personal use.

- You may download it free from the R/C Soaring UK web site at http://www.rc-soar.com
- You may make copies for other people, as long as it is for their own personal use. Please copy the original in its entirety without any changes.
- If you receive a copy of this guide from somebody else, you also agree to abide by the terms of the licence.

What you may not do:

- You may not sell copies at a profit.
- You may not publish it in whole or in part electronically (e.g. via a web site) without express permission.
- If you are a retailer, this licence does not enable you to distribute copies to customers. If you wish to do this please contact me at mpx3030@rc-soar.com

## Acknowledgements

First and foremost, thanks first to Multiplex USA (http://www.multiplexrc.com) for their invaluable support. If you are looking to buy MPX equipment or accessories, please take a look at their Dealer Locator for all the Americas at http://www.multiplexrc.com/dealers.htm.

Thanks also to Multiplex for providing the LCD font for the screen shots, Harry Curzon for his contributions, and to those modellers who kindly provided feedback on the draft versions.

## How to Use This Guide

Terminology

I have changed some of the terms used in the user manual. For example “options” are now called “attributes”, “lists” are called “model memories” and so on. This is done to reflect common usage, or to improve clarity.

Typographical Conventions

The following typographical conventions are used:

- Programming screens are **bold** such as when you go to the **Assign Servos** screen.
- Control attributes are **BOLD CAPS** such as **EXPO**, **DIFFER** and **DUALRATE**
- Mixers are upper case e.g. BUTTERFL

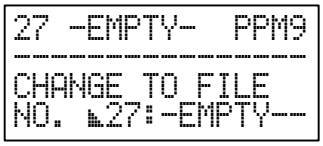
# Quick Start

Let’s jump straight in with a simple example to get the feel of programming the unit and gain some idea of the MPX 3030 software architecture. The goal is simple: to control a single servo on channel 1 via the right hand stick.

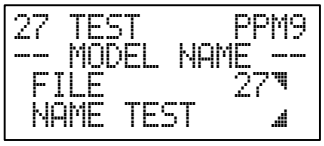
You will need to be familiar with basic keypad and menu operation, if not, then take time off to read the, *Keypad and Menu System* on page 10 of the mc3030 manual. To learn more about the individual screens, refer to *Appendix H - Program Screen* in this guide.

Follow these steps:

1. Go to the **File Shift** screen, select an empty memory, i.e. one named “-EMPTY--“




2. Go to the **File Name** screen, change the name from “-EMPTY--” to “TEST”.




3. Go to the **Assign Control** screen, assign Aileron to “C”.


4. Go to the **Assign Servo** screen, assign Servo 1 to Aileron.

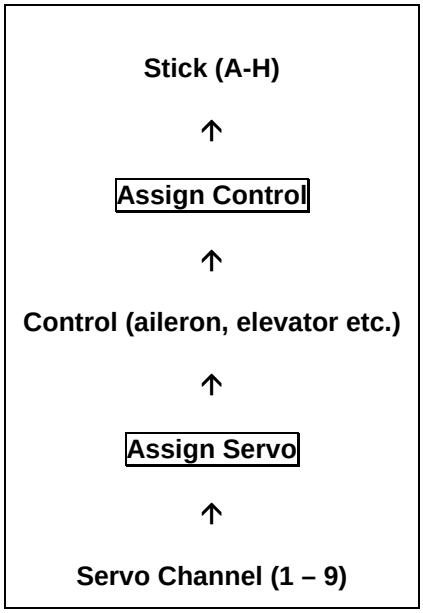
5. Press M until you get back to the opening screen.
6. Plug a servo into channel 1, switch on the transmitter and receiver, and move the right hand stick from side to side. The servo should follow the stick.

Note that there were three "things" involved in the programming process, namely

- Sticks
- Controls
- Servos

These can be represented as three layers with *sticks and switches* at the top, *servos* at the bottom and *controls* in the middle. For a servo to function there must be a route from servo to stick. We can show this diagrammatically:

**Figure 1**
**Assigning Controls and Servos**




As you can see from Fig. 1, the **Assign Controls** screen is used to create the link between controls and sticks. Similarly the **Assign Servos** screen links servos to controls.

Notice the direction of the arrows. The **Assign Servo** screen assigns a servo to a control (not the other way round), so the arrow is shown pointing upwards. Same applies to **Assign Control** – a control is assigned to a stick so again the arrow is an upward pointing one.

When you set up a new configuration you will *always* start with the **Assign Control** and **Assign Servo** screens, in that order. *These are the two most important screens in the system!* Note that there are also two corresponding "Adjust" screens, namely **Adjust Control** and **Adjust Servo**. These are used only after the control and servo assignments. It's a little confusing because the Assign screens are at the Menu 2 level even though they are the first ones you'll be using to set up your model. The rationale behind this is that although you use the Assign screens first, you will probably end up using the Adjust screens more often during day to day flying.

# Widgets, Controls and Servos

Now you've looked at a simple example, let's take a detailed look at the objects involved in programming the transmitter, namely Widgets, Controls, Servos and Mixers.

## Widgets and Ports

*Widgets* are things which the user can move on the transmitter front panel, i.e. the sticks, switches, sliders and knobs. I've called them widgets because it's more convenient than "sticks, switches, sliders and knobs" although I will occasionally slip into the old terminology!

Each widget has a wiring harness which plugs into a *port* on the main board inside the transmitter. Each port has a code embossed next to it. The codes are used in the various programming screens to identify the individual sticks and switches.

The codes A-G displayed on the front of the transmitter correspond to the default wiring of the sticks and switches. If you connect the widgets to different pots inside the transmitter, the legends will cease to be correct. Fortunately it's easy to identify the port that a switch or slider is connected to internally by using the **Test Widgets** screen.

Ports fall into three categories:

1. The nine ports A-I are reserved for *primary widgets* i.e. the sticks and switches which provide the primary functions. A, B, C and D are reserved for the twin axis sticks (one code for each stick *axis*, that's two per stick *unit*)[1]. Ports E, F, G, H and I are for sliders and switches.

2. Ports S1–S5 and LS are for the *secondary switches.* Note that the manual uses the term "changeover/coupling switches" but this is a bit of a mouthful so I've shortened it – think of S as short for "secondary". Secondary switches are used for a variety of auxiliary functions described later. They cannot be used to drive servos directly.[2] LS is the pupil-teacher ("buddy box") switch but can also be used as a secondary switch if a buddy box is not in use.

3. The remaining ports are DE for the Digi Adjuster and M for the Model Memory switch.

The majority of this guide is concerned with the primary and secondary widgets (the first two groups above). For more details, see *Appendix A - Mainboard Connections*.

# Controls

### *Personalities and Controls*

Advanced computer radios recognise that different channels are used for different purposes and require different adjustments. For example, ailerons often have differential adjustment to help with axial rolls, while idle trim on the throttle allows the low throttle position to be adjusted without altering the top end setting. It could be said that aileron and throttle functions have distinct *personalities.* The 3030 uses the concept of *controls* to represent these personalities. Controls have easily remembered names such as "Aileron", "Elevator" and "Rudder".

### *Control Attributes*

Each control in the mc3030 has its own set of distinctive adjustable *attributes*[3]*.* For example the Aileron control has **EXPO**, **DUAL RATE**, **TRAVEL**, **CENTRE** and **DIFFER** attributes; while the Spoiler control has **FIX.VAL.1**, **IDLETRIM** and **NORMPOS** attributes.


Attributes are adjusted in the **Control Setup** menu. Most attributes have an associated value e.g. the **DIFFER** is adjustable between 0 (no differential) and 100 (maximum).

Some attributes appear in several controls, e.g. **TRAVEL+/-**

[1] The stick units have five pin plugs. They are labelled KnL and KnR respectively. KnL corresponds to ports A and B, KnR to C and D.

[2] Unless used with the FIX VAL virtual control.

[3] Note that the use of the term *attributes* is taken from computer programming practice – they are called "options" in the user manual.

applies to all the main controls. Others are specific to particular controls, e.g. **THR.CURV** is an attribute of the Pitch control, for helicopters.

### *Assigning Controls to Sticks*

As you saw in the *Quick Start* example, there is no such thing as a ready made "throttle stick" or "flap switch" on the mc3030. To make a throttle stick, then you must assign the Throttle control to stick B.

By the same token, if you want the left hand stick to control your airbrakes, then assign Spoiler to stick B. In other words, always *choose the correct control* for the task then *assign it to your stick of choice*. If you use an inappropriate control you'll see inappropriate attributes in the **Control Setup** menu, and the predefined mixers may not work as you expect.

### *Control Categories*

Controls fall into four broad categories for Aircraft, Boats, and Helicopters, plus the four Aux controls for simple applications. For a basic control without bells and whistles, consider using one of Aux 1–4. These have the basic **TRAVEL**, **CENTRE** and **FIX.VAL.1** attributes.

The programming screens for Controls are

- Assign Controls
- Control Setup

For a complete list of controls and attributes see *Appendix B - Controls and Attributes*. To adjust attributes see *Adjusting the Transmitter Controls* on pages 37-43 of the user manual.

## Servos and Channel Numbers

A servo is known to the system by its channel number 1 to 9. A servo channel is activated simply by assigning the servo to a control (or mixer) in the **Assign Servo** screen. Actual channel numbers are unimportant[4].

As an aside, flexibility in channel numbering is one of the strengths of the 3030, since you never need more channels than you have servos. This allows you to use smaller, cheaper receivers even for applications with sophisticated mixing.

The following screens are used for setting up servos and channels.

- Assign Servo
- Servo Travel
- Servo Centre
- Servo Limit

For a complete list of controls and mixers see *Appendix E - Servo Assignment Targets*.

# Mixers

In this section we'll take a brief look at mixers in general. Later on, we'll look at user mixers in more detail.

[4] A particular servo numbering order must be observed for wing servos when using the BUTTERFL and QUADRO mixers – see *User Manual for the Profi mc 3030* p.81.

## What are Mixers?

We have seen that in order to make a servo respond to a transmitter stick, you must assign a servo to a control. But what happens if a servo is driven by more than one control – e.g. a V tail servo responds to both rudder and elevator commands? We have a problem, because the **Assign Servo** screen will only allow you to assign a servo to a single control.

However, it will also allow you assign a servo to a *mixer*. A mixer is nothing more than a list of two or more controls, conveniently bundled together and given a name. For example The V.TAIL mixer is just a list of two controls, Rudder and Elevator.

The way it all works as follows: when you assign a servo to a mixer and press the **M** key in the **Assign Servo** screen, the system looks at the mixer's controls, and sets up a page in the **Servo Travel** screen for each control. When you go to the **Servo Travel** screen, you can use each "page" to adjust the movement of the servo for the corresponding control. This is called *adjusting the servo mix* and the mixer's controls are commonly referred to as servo's *mixer inputs*.

For example, if you assign a servo to the V.TAIL mixer, when you press the **M** key two pages are created in the **Servo Travel** screen for the servo, one each for Rudder and Elevator controls.

## When do You Need to Use a Mixer?

As we have seen, a mixer is required *if the position of a servo is determined by more than one control*. This is very important to understanding the MPX programming system and we'll make use of this in the 3-step programming procedures you'll be using. Mixers can be categorised into two types: built-in and user-defined.

## Built-in Mixers

There are thirteen built-in mixers for specific applications. They have easily remembered names like DELTA and V.TAIL. For a complete list see *Appendix G - Predefined Mixers*.

## User Mixers

For special requirements where built-in mixers won't do, you can create *user-defined* mixers. We'll look at these in more detail later on. The following table lists the differences between built-in and user-mixers.

**Table 1 – Comparison of built-in and user- mixers**

| Mixer Type | Can change inputs? | Can change trim mode? | Set-up Procedure |
|---|---|---|---|
| Built In Mixer | No | No | 1. Assign servo(s) to mixer in **Assign Servos**<br>2. Adjust mixing in **Servo Travel** |
| User Mixer | Any controls, max 4. | Yes | 1. Create a USR1-3 mixer in **User Mix** screen.<br>2. Assign servo(s) to mixer in **Assign Servos**<br>3. Adjust mixing in **Servo Travel** |

## Mixer "Outputs"

Although mixers all have between two and four *inputs*, there is no concept of mixer *outputs*. To use a mixer you simply assign a servo to the mixer in the **Assign Servo** screen. The

additional pages for the mixer adjustments will automatically appear when you next go to the **Servo Travel** screen.

# Configuring a Model from Scratch

We've looked at the basic architecture of the MPX. Now it's time to look at a set of procedures for programming the system from scratch.

At this stage we're only interested in the basic operation, i.e. getting servos to respond correctly to the sticks.

The procedures are completely general: although the examples are based on gliders, they can be applied to model of varying types and complexity, and you will see the same steps repeated all the way up to the complex F3F example.

The steps involved in basic set-up are as follows:

**STEP ONE:** **Assign controls** - specify sticks and sliders to be used for primary controls.

**STEP TWO:** **Assign servos** – assign servos to controls and/or mixers.

**STEP THREE:** **Adjust servo centres, travel and mixing**.

# EXAMPLE 1: V-tail Soarer

We'll use a V-tail glider as the subject of our first example.

- The model has two servos driving the V-tail and two servos driving the ailerons.
- The V-tail functions as a both a Rudder and Elevator.

## Step One: Assign Controls

The first step is to decide which sticks and widgets are to use. This information is needed for the **Assign Controls** screen which is the first one you'll use. Don't concern yourself about the servos at this stage or about secondary functions like Dual Rates etc – you should be concerned just with the controls as listed in *Appendix B - Controls and Attributes*. For our example a typical assignment would be as follows:

**Table 2 - Control Assignments for V-tail Glider**

| Control | Widget |
|---|---|
| Rudder | A (left stick left/right) |
| Aileron | C (right stick left/right) |
| Elevator | D (right stick up/down) |

You can now invoke the **Assign Control** screen to set up these assignments. Make sure unused widgets are set to UNUSED. Here are the three assignments as they appear in the **Assign Control** screen:




## Step Two: Assign Servos

You've linked the controls to the widgets. It's now time to think about the other part of the process, linking your servos to the controls. This is the most complex part of the process as it may involve choosing a mixer, so it's broken down into a further set of steps.

### *Create the Servo Assignment table*

Start by making an empty table like Table 3, with one row per servo. The columns are as follows:

- **Servo** – description and location of the servo e.g. "V-Tail left".
- **Channel** – the channel number 1 – 9.
- **Controls** – a list of one or more controls affecting the servo. All the controls in Table 2 will be listed at least once somewhere in this column.
- **Servo Assigned To** – the control or mixer that the servo is assigned to in the **Assign Servo** screen.
- **Servo Travel Pages** – a list of one or more "pages" which will appear in the **Servo Travel** screen for the servo.

### *Specify Channel Numbers*

Enter the servo descriptions and their channel numbers into the table. If you're using a miniature receiver with four or five channel outputs, you'll want to use the low channel numbers.

**Table 3 –servo assignment table with channel numbers**

| Servo | Channel | Controls | Servo assigned to | Servo Travel Pages |
|---|---|---|---|---|
| V-tail LEFT | 1 | | | |
| V-tail RIGHT | 2 | | | |
| Aileron LEFT | 3 | | | |
| Aileron RIGHT | 4 | | | |

### *List Controls*

You now have to list the controls which will affect each servo. The easiest way is to imagine you're sitting in front of your model with the transmitter on, waggling the controls. Imagine which servos move when you activate each control in Table 2, and make corresponding entries in the Controls column.

In our V-Tail example, displacing the elevator stick will move servos 1 and 2, so enter "Elevator" in the Controls column of channels 1 and 2. Displacing the rudder stick also moves servos 1 and 2, so add "Rudder" as well to channels 1 and 2.

When you move the aileron stick, only the aileron servos move, so enter "Aileron" in the Controls column of channels 3 and 4. The table ends up like this:

**Table 4 – servo assignment with controls**

| Servo | Channel | Controls | Servo assigned to | Servo Travel Pages |
|---|---|---|---|---|
| V-tail LEFT | 1 | Rudder, Elevator | | |
| V-tail RIGHT | 2 | Rudder, Elevator | | |
| Aileron LEFT | 3 | Aileron | | |
| Aileron RIGHT | 4 | Aileron | | |

***Choose Mixers***

Now we need to determine the controls and mixers to use in the **Assign Servos** screen. A servo can be assigned to either a control or a mixer. The rule is:

- If one control is specified, the servo is assigned directly to the control.
- If more than one control are specified, assign the servo to a suitable mixer.

In our example, the two aileron servos on channels 3 and 4 are driven solely by the Aileron control, so no mixer is required for these. We therefore enter "Aileron" in the fourth column.

However, servos 1 and 2 are each driven by both rudder and elevator controls so we cannot assign each directly to a single control – we must assign them to a mixer.

To determine which mixer to use, turn to *Appendix G - Predefined Mixers* and look for a mixer which (a) has at least Rudder and Elevator as inputs, and (b) looks like it is designed for the task!

Both the V.TAIL and V.TAIL+ mixers are suitable candidates. Note that the V.TAIL+ mix has Spoiler and Flap inputs in addition to Rudder and Elevator. Let's use the plain V.TAIL mixer first. Here is the assignment table with the addition of the VTAIL mixer:

**Table 5 – Servo Assignment Table (cont.)**

| Servo | Channel | Controls | Servo assigned to | Servo Travel Pages |
|---|---|---|---|---|
| V-tail LEFT | 1 | Rudder, Elevator | V.TAIL (mixer) | |
| V-tail RIGHT | 2 | Rudder, Elevator | V.TAIL (mixer) | |
| Aileron LEFT | 3 | Aileron | AILERON | |
| Aileron RIGHT | 4 | Aileron | AILERON | |

### *List Servo Travel Pages*

In this step you list the pages which will appear in the **Servo Travel** screen.

- If the servo is assigned to a *mixer* in column 4, enter the names of *all* the inputs to the mixer into column 5. The complete list of inputs is in *Appendix G - Predefined Mixers.* Unused inputs must have the operating mode set to OFF.
- If the servo is assigned to a *control* in column 4, copy the name of the control into column 5.

We end up with the following table:

**Table 6 – Completed servo assignment using V.TAIL mixer**

| Servo | Channel | Controls | Servo assigned to | Servo Travel Pages |
|---|---|---|---|---|
| V-tail LEFT | 1 | Rudder Elevator | V.TAIL | Rudder, Elevator |
| V-tail RIGHT | 2 | Rudder Elevator | V.TAIL | Rudder, Elevator |
| Aileron LEFT | 3 | Aileron | AILERON | Aileron |
| Aileron RIGHT | 4 | Aileron | AILERON | Aileron |

What if we envisaged using Spoiler control at a later date? In that case we'd be better off starting with the V.TAIL+ mixer. Note that I've marked Flap and Spoiler inputs as "off", in fact in this particular example Flap and Spoiler controls are not used so these mixer inputs are inactive anyway.

**Table 7 – Completed servo assignment using V.TAIL+ mixer**

| Servo | Channel | Controls | Servo assigned to | Servo Travel Pages |
|---|---|---|---|---|
| V-tail LEFT | 1 | Rudder Elevator | V.TAIL+ | Rudder, Elevator, Spoiler(OFF), Flap(OFF) |
| V-tail RIGHT | 2 | Rudder Elevator | V.TAIL+ | Rudder, Elevator, Spoiler(OFF), Flap(OFF) |
| Aileron LEFT | 3 | Aileron | AILERON | Aileron |
| Aileron RIGHT | 4 | Aileron | AILERON | Aileron |

The servo assignments are now complete, so go to the **Assign Servos** screen, assign Channels 1 and 2 to V.TAIL (or V.TAIL+):



Assign channels 3 and 4 to AILERON:


That's the end of the Step Two – we now have now completed the routing of servos to controls to sticks and switches.

## Step Three: Adjusting Servo Travel and Centre

Now it's time to get the model out and adjust the servos!

Use the **Servo Centre** screen to adjust the centre offset of each servo.

Next, use the **Servo Travel** to adjust the servo travel and mixing. Each servo has one page for each inputs affecting that servo – these are the ones you identified in the "servo travel pages" column of Table 6 and Table 7. The example screen shots show the pages for elevator and rudder mixer inputs for servo 1.



These page shows that (a) pulling back on the elevator stick D causes 30% movement on servo 1, and (b) moving rudder stick A full left generates 30% movement in the same servo. Each of these inputs is currently ON, although later we'll see how they can be controlled via a secondary switch.

If an input is unassigned, the system displays "CONTROL?" along the bottom line. In our example, the V.TAIL+ mixer has Flap and Spoiler inputs which are unassigned - in this case because we're not using them, not because we forgot to assign them!


For more information refer to:

- Appendix H - Program Screens (p.45)
- Adjusting Servo Travel (p.16)
- Adjusting Servo Centre (p.17)

# EXAMPLE 2: 60" Pylon Racer

Let's look at a real world example, that of a 60" pylon racer class model. A typical example would have the following features:

- Two servos in the wing, one for each aileron.
- Two servos in the fuselage operating the V tail.
- "Aileron brakes". This is a simplified version of true crow brakes for models which do not have flaps. When spoiler is applied, both ailerons move upwards together[5].
- Spoiler→Elevator mixing to compensate for the pitch-up generated by crow brakes.

After you've done the basic set-up for the above, you'll see how to add:

- Snapflap (Elevator→Aileron mixing)
- Coupled Ailerons and Rudder (optional)

How do you implement such a seemingly complex configuration? The answer is "exactly the same as before" - there's just a little more data entry involved!

## Step One: Assign Controls.

The first example had three controls Rudder, Elevator and Aileron. Our pylon racer also has simple crow brakes operated from the left-hand stick. Spoiler is the most appropriate control for crow brakes because there is a special BUTTERFL mixer for this purpose[6]. And since we will use the left-hand stick to operate spoiler, the modified control assignment is as follows:

**Table 8 - Control Assignments with Addition of Spoiler**

| Control | Port (widget) |
|---|---|
| Rudder | A (left stick left/right) |
| Aileron | C (right stick left/right) |
| Elevator | D (right stick up/down) |
| Spoiler | B (left stick up/down) |

## Step Two: Assign Servos

The servo assignment is similar to *Example 1*, except for the addition of spoiler control. Here are the steps in detail.

### *List Controls*

Let's take the *Example 1* as a basis and see the effect of adding a spoiler control. As before imagine you're controlling the model, moving the Spoiler stick.

What is the effect on the servos? Moving the spoiler stick will cause both ailerons to move upwards. At the same time we want both V-tail surfaces to move down to compensate for the

[5] Crow brakes normally use separate flaps and ailerons, but the flaps can be dispensed with for smaller models.

[6] "Butterfly" and "crow" mixing are synonymous.

resulting nose-up trim change. Therefore all four servos are affected by moving the Spoiler stick, so we need to add Spoiler to the list of controls for every servo. Here is the Servo Assignment table modified to take into account the Spoiler control:

**Table 9 – Servo Assignment with Spoiler Control**

| Servo | Channel | Controls | Servo Assigned to | Servo Travel Pages |
|---|---|---|---|---|
| V-tail LEFT | 1 | Rudder, Elevator, Spoiler | | |
| V-tail RIGHT | 2 | Rudder, Elevator, Spoiler | | |
| Aileron LEFT | 3 | Aileron, Spoiler | | |
| Aileron RIGHT | 4 | Aileron, Spoiler | | |

***Choose Mixers***

To recap, for each servo,

- If a single control is specified against a servo, assign the servo directly to that control.
- If two or more controls are specified, assign the servo to a suitable mixer.

In this case, *all* the servos must be assigned to mixers since each box in the Controls column contains two or more entries. There are two groups of inputs – Rudder + Elevator + Spoiler for the V-tail servos, and Aileron + Spoiler for the Aileron servos.

Scanning through the definitions of the predefined mixers shows you will see that we can use two mixers, BUTTERFL for Aileron and Spoiler inputs and VTAIL+ for Rudder, Elevator and Spoiler inputs. Note that with this scheme, the Spoiler control is an input to two mixers – that is perfectly valid.

Having chosen the mixers, we can now enter them in column 4 of the servo assignment table:

**Table 10 – Servo Assignment Table (cont.)**

| Servo | Channel | Controls | Servo Assigned to | |
|---|---|---|---|---|
| V-tail LEFT | 1 | Rudder, Elevator, Spoiler | V.TAIL+ | |
| V-tail RIGHT | 2 | Rudder, Elevator, Spoiler | V.TAIL+ | |
| Aileron LEFT | 3 | Aileron, Spoiler | BUTTERFL | |
| Aileron RIGHT | 4 | Aileron, Spoiler | BUTTERFL | |

***List Servo Travel Pages***

To recap: in this step you complete the last column showing what pages will appear in the **Servo Travel** screen. Again there are two possibilities depending on whether the servo is assigned to a mixer or directly to a control.

- If the servo is assigned to a *mixer* in column 4, enter the names of *all* the inputs to the mixer into column 5. To get the complete list of inputs, see *Appendix G - Predefined Mixers*. Unused inputs must be disabled, i.e. have the operating mode set to OFF.
- If the servo is assigned to a *control* in column 4, copy the name of the control into column 5.

In this example *all* the servos are assigned to mixers rather than controls.

**Table 11 – Completed servo assignment table**

| Servo | Channel | Controls | Servo Assigned to | Servo Travel Pages |
|---|---|---|---|---|
| V-tail LEFT | 1 | Rudder, Elevator, Spoiler | V.TAIL+ | Rudder, Elevator, Spoiler, Flap(OFF) |
| V-tail RIGHT | 2 | Rudder, Elevator, Spoiler | V.TAIL+ | Rudder, Elevator, Spoiler, Flap(OFF) |
| Aileron LEFT | 3 | Aileron, Spoiler | BUTTERFL | Aileron, Spoiler, Elevator(OFF), Flap(OFF) |
| Aileron RIGHT | 4 | Aileron, Spoiler | BUTTERFL | Aileron, Spoiler, Elevator(OFF), Flap(OFF) |

## Step Three: Adjusting Servo Travel and Centre

Note that in this example, there is no flap control used (i.e. the Flap has not been assigned to a stick). So any Flap inputs to the V.TAIL+ mixer are ignored in the servo travel pages.

However, the Elevator control *is* assigned to a stick, so remember to set the mode to OFF for the elevator pages for servos 3 and 4.

## Extra Steps: Adding Bells and Whistles

We've seen how to do the basic set-up for our pylon racer model. Let's see how to soup things a bit – hopefully without needing to completely re-programme the model. After all you don't want to have to completely reconfigure the system just to add a little bit more mixing!

### *Adding Elevator à Aileron mixing ("snap-flap")*

Now let's add snap-flap mixing.[7] 60" racers do not generally have separate flaps, but it's quite common in such models to use the ailerons as flaps for the purposes of snap-flap.

The changes are quite straightforward.

Let's look at the servo assignment table again and do the usual trick of pretending to control the model. Now when we pull back on the elevator stick, the ailerons droop slightly, so we must add Elevator to the list of controls for the aileron servos.

**Table 12 – Servo assignments after addition of snap-flap mixing**

| Servo | Channel | Controls | Servo Assigned to | Servo Travel Pages |
|---|---|---|---|---|
| V-tail LEFT | 1 | Rudder, Elevator, Spoiler | V.TAIL+ | Rudder, Elevator, Spoiler, Flap(OFF) |
| V-tail RIGHT | 2 | Rudder, Elevator, Spoiler | V.TAIL+ | Rudder, Elevator, Spoiler, Flap(OFF) |
| Aileron LEFT | 3 | Aileron, Spoiler, Elevator | BUTTERFL | Aileron, Spoiler, Elevator(S1), Flap(OFF) |
| Aileron RIGHT | 4 | Aileron, Spoiler, Elevator | BUTTERFL | Aileron, Spoiler, Elevator(S1), Flap(OFF) |

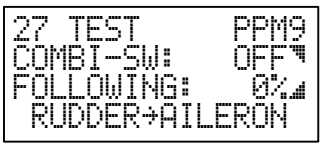
You also now need to check that the mixer for channels 3 and 4 can handle the extra Elevator control. Fortunately BUTTERFL can handle elevator inputs – no great surprise as it was designed for crow brake applications.

[7] Snap-flap i.e. elevator→flap mixing is often used for aerobatic and F3F soarers.

Another little change: since elevator is now used as input to the BUTTERFL mixer, the “OFF” qualification against Elevator in the last column has been replaced with “S1” – this means snap flap mixing can be switched on and off using switch S1. This is jumping ahead a bit - we’ll learn more about the use of secondary switches for this purpose later - this technique for disabling specific mixer inputs in flight is often of use.

You’ll see that the Flap control is disabled in all the servo travel pages. In fact you don’t need to do this explicitly since the Flap control is not used on this model, however you can see that adding a flap control for a full-house F3F or F3B machine should mean very little extra work!

#### *Adding Coupled Ailerons and Rudder*




Let’s add coupled ailerons and rudder (“CAR”). This is set in a special screen as shown at left. It can be added retrospectively to an existing model set-up. In fact you don’t need to consider rudder and aileron coupling at all when creating the Servo Assignment table for a new model, just configure it assuming no CAR, then add it later. For more information see *Coupled Ailerons and Rudder* (p.23).

# More about Servo and Control Adjustments

Let’s take a step back and look at some tips and tricks to make life simpler when setting up the model. As with many computer radios, there are often several ways of accomplishing the same goal. We’ll now focus on the different ways of adjusting servo travel and centring.

## Adjusting Servo Travel

There are two ways to adjust servo travel.

#### *Using the Servo Travel Screen*

The **Servo Travel** screen operates on individual servos. Use this screen to set the servo travel on the bench before the first flight.

#### *Using the TRAVEL Attribute*

The **TRAVEL+/-** and **TRAVEL** attributes affect the travel of *all* servos affected by a control both directly or via mixers. For example if you are using a BUTTERFL mixer and you increase the **TRAVEL** of the aileron control, it will increase the travel of both ailerons and flaps.

**TRAVEL+/-** is an attribute of Elevator, Rudder, Aux and some of the heli controls and allows travel adjustment in each direction. **TRAVEL** (i.e. without the +/-) is used for the Aileron control and adjusts travel in both directions together – this ensures that differential settings are preserved.

## Adjusting Servo Centres

As with servo travel adjustment, there are two ways to adjust servo centres.

#### *Using the Servo Centre Screen*

The **Servo Centre** screen is used to centre individual servos, during initial set-up of the model or when readjusting due to a bent linkage. Adjust each servo till the centre is perfectly set using whatever reference is necessary, e.g. control surfaces lining up etc.

#### *Using the CENTRE Attribute*

To adjust the *trim* centres, adjust the **CENTRE** attribute of the control. This affects *all* servos affected by a control either directly or via a mixer. You can use **CENTRE** to re-centre your trim levers following a flight trimming session. The way this works is as follows: when you displace the trim lever, the system alters the value of the **CENTRE** attribute to reflect the amount of trim which has been applied. You can see this happening in real time as follows:

**7.** Go into **Control Setup,** select the Aileron control and the **CENTRE** attribute.

**8.** Move the Aileron trim slider and watch **CENTRE** value change.

If you you've made some trim adjustments using the trim lever, and want to move a trim lever back to the centre detent,

9. Note the value of the **CENTRE** attribute.

10. Move the trim lever back to the centre detent. Watch the value of the **CENTRE** attribute change as you do this!

11. Restore **CENTRE** to the value in (1) above using the Digi Adjuster or +/- keys.

Note that not all controls have a **CENTRE** attribute. See *Appendix B - Controls and Attributes* for the complete list.

# Adding Spice: Secondary Switches

Let's recap: so far you've learnt how to configure your sticks, you've got them to move the servos correctly, the mixers are properly configured, the servos are centred and they have the correct travel in both directions – every thing is working in a nice linear fashion. In fact at this point you could quite happily go out and fly the model.

However with many models there will be situations where you don't want things to work in a nice linear fashion. You may want to switch certain functions off, or over-ride a particular proportional function (e.g. suppress spoiler operation) or send a servo to a fixed position (for flaps). This is where secondary switches come in handy.

Secondary switches are used in a number of ways in the MPX 3030

(1) To activate mixer inputs.

(2) To activate the **DUALRATE**, **FIX.VAL.1** and **FIX.VAL.2** control attributes.

(3) To activate CAR (Coupled Ailerons and Rudder).

(4) To trigger the Timer function.

(5) To operate the FIX VAL virtual control.

Secondary switches are implemented in three different ways which we'll look at in the following sections.

## Secondary Switches S1–5, L/S

Ports S1-5 and L/S use 2-position switches. They can present two states (on/off) to the system software.

## Secondary Switch "SI"

We have already seen the port I can be used for primary controls just like A-H. Alternatively it can be used either as a secondary switch where it is identified as "SI" in the screens. (note "ess eye", not "ess one"!).

SI is different to the two position switches S1-5, because it can present three states to the system software. It therefore requires a 3-position switch in this role. If you haven't got a spare 3-position switch then you can use the 3-position switch supplied for G – remove the back of the transmitter, disconnect the plug in port G, and reconnect it to port I.

The three states are:

- Switch at one side: ↑
- Switch at other side: ↓
- Switch in the middle: Off.

The system software uses the Off state of to provide a second preset position for Flap, via the FIX.VAL.2 attribute. There are other undocumented uses of the Off state which we'll see later.

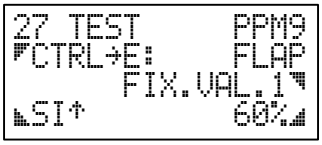
***Using SI for 2 preset flap positions.***

SI provides a second preset position for the flap control – this feature is *only* available via SI and *only* if a 3-position switch is plugged into port "I". Here's how you do it:

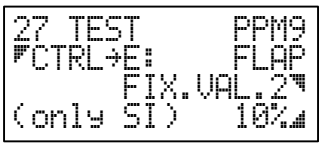
1. Go into **Control Setup** screen, and select Flap, and move to the **FIX.VAL.1** attribute.

2. Set the value for **FIX.VAL.1** and set the mode to SI. This sets the first preset position.

3. Change the mode from Off to SI↑. At this stage the screen will look something like:



4. Move to the **FIX.VAL.2** attribute. The menu displays “FIX.VAL.2 (only SI)”. Set the value for **FIX.VAL.2.** This sets the second preset position:




With SI at either side of its travel, the two preset positions of 10% and 60% kick in. With SI in the centre position the flap is controlled as normal by the slider it’s assigned to.

## Secondary switch “Gx“

Gx is a “virtual” secondary switch, i.e. it’s a software switch, not a physical switch like S1-5. However it can be used anywhere S1–5 are used. It is triggered by moving a primary stick A-H past a user-definable position.

Typical uses are to trigger the Timer function from the towhook release control, or to switch dual-rate setting as the throttle stick is moved from low to high, or to disable snap-flap mixer inputs when flaps are applied.

In the **Soft Switch** screen, you assign the “master” widget and the trigger point. Afterwards, wherever you can use a secondary switch you’ll see an additional choice in the secondary switch menus, called SA, SB, SC, SD, SE, SF or SG depending on the master widget.

## Choosing the Correct Type of Secondary Switch

As you’ve seen there are three different types of secondary switch, S1-S5, SI, and Gx. How do you choose the right one for a particular application?

- If you want a switch to set two preset positions for flaps, SI is the only choice.
- For other cases choose S1-S5 if you want to flick a switch, or Gx if you want to trigger the function via from another control.

# Setting up Ailerons, V-tails and Spoilers

## Setting up Ailerons

The mc3030 allows easy adjustment of differential and control sensitivity, with a single point of adjustment for each, but to take advantage of this you need to be systematic in the way you set up the model.

1. When setting up on the bench, turn off **DIFFER** (by setting to zero, or disabling it), and use the **Servo Travel** screen to obtain maximum servo travel (subject to the mechanical limits of the hinges) and make sure there is *equal* movement in both directions.

2. If differential is required, simply dial in a suitable figure say 50% for the Aileron’s **DIFFER** attribute using the **Control Setup** screen for Aileron. The downgoing movement servos will now be ½ the movement of the upgoing movement. If you find that the upgoing aileron is affected instead of the downgoing, reverse the **DIFFER** value.

3. During flight trials, you can reduce the control movement of *all* the aileron servos together, using the **Control Setup** screen and the **TRAVEL** attribute. Or you can change the differential for all servos at once simply by altering the **DIFFER** attribute[8].

## Limiting Servo Travel

We've already seen in the examples how to set up V-tail mixing. One further consideration is the effect of applying rudder and elevator together. If the individual controls have a large amount of movement, then it's possible that applying both rudder and elevator together will cause one control surface to try to exceed the mechanical limits of the hinge. To prevent this, there are three options:

1. Reduce the value of the **TRAVEL** attribute for each control.

2. Reduce the travel of individual servos, using the **Servo Travel** screen.

3. Use the **Servo Limit** screen to limit servo movement.

The first two methods are the least satisfactory as they reduce the sensitivity of the individual controls. The third method preserves sensitivity, but bear in mind that control surface movements will be non-linear when both inputs are at a maximum.

## Setting up Spoilers

Spoilers are supported in two ways on the mc3030 (a) by the Spoiler controls attributes, and (b) by three special mixers.

### *The Spoiler Attributes*

There are many different types of design of spoiler: letter-box, scissor airbrakes, crow brakes (2- and 4-servos) and canopy brakes spring to mind. Fortunately the Spoiler attributes **NORMPOS** and **IDLETRIM** have been design to deal with all these variants so you can use the Spoiler control for any of these applications. Let's look at these attributes in more detail.

#### NORMPOS

You set the **NORMPOS** attribute to indicate to the system software the position of your stick for "spoiler closed". Unlike other controls there is no convention for the direction of operation of the spoiler stick or slider – some pilots have forward-to-open, others back-to-open. The system must have this information in order to calculate how far the spoiler is displaced from the "closed" position. In practice there is a bug in v3.0! We'll come to that later.

#### IDLETRIM

With letterbox and scissor-action spoilers, it's often desirable to be able to fine adjust the "spoiler closed" side of the servo travel without affecting the "open" side. This lop-sided trim function is called Idle Trim, a term borrowed from power flyers who need to do the same sort of thing with Throttle – in fact Idle Trim is also available on the Throttle control.

If you are using stick B or D to operate spoiler you can use the **Idle Trim** attribute to control the maximum trim range at idle. The larger the value of **Idle Trim**, the greater the trim range in the "closed" side. Whatever value is set, the trim range decreases progressively to zero as

[8] You can even do this in flight using the DigiAdjuster, but remember to close the flap to the programming panel first!

the spoiler is opened so that at fully open, displacing the trim lever has no effect. Note Idle trim does not operate with the BUTTERFL mixer as neutral adjustment is handled via the aileron and flap trim levers.

#### *Mixers for Spoilers*

In addition to the Spoiler attributes, there are three built-in mixers designed for use with the Spoiler control. These are called BUTTERFL, ELEV MIX, and V.TAIL+.

### The BUTTERFL Mixer (and the NORMPOS Bug!)

The BUTTERFL mixer is used to implement crow-brake and snapflap (elevator➔flap) mixing for F3x models. You assign all the wing servos to the BUTTERFL mixer. As we have seen in the 60" racer example, it can also be used for simple "ailerons up" braking in 2-servos-in-wing models by disabling the Flap input.

The BUTTERFL mixer also provides aileron differential suppression: as full crow is applied, any aileron differential is gradually reduced until at full spoiler the differential is zero. This provides better aileron control when crow brakes are deployed. However because of a bug in the software, it only works properly if **NORMPOS = ↑**. If you set **NORMPOS = ↓,** aileron differential is suppressed when the brakes are *closed*, which is the opposite of what's required.

### The ELEV and V.TAIL+ Mixers

A side effect of applying crow brakes is to apply a nose-up pitching moment. To compensate for this you will need Spoiler➔Elevator mixing so that some down elevator movement is applied as brakes are deployed. For a V-tail model assign the V-tail servos to the V.TAIL+ mixer. For a conventional tail model, assign the elevator servo to ELEV MIX. Each of these mixers has Spoiler and Elevator inputs.

In fact, a common configuration for F3F, F3J and F3B models is to use BUTTERFL together with either ELEV MIX or V.TAIL+.

#### *Scissor, Letterbox and Canopy brakes*

Simple brakes such as scissor, letterbox and canopy brakes can be set up without complex mixing:

1. Assign the Spoiler control to a suitable widget, normally the left hand (ratchet) stick.

2. Assign the airbrake servo to the Spoiler control.

3. If you want mix spoiler➔elevator for elevator trim compensation, then

- for a conventional tail model, assign the Elevator servo to the ELEV MIX mixer.
- for a V-tail model, assign the V tail servos to the V.TAIL+ mixer.

4. To adjust the spoiler-closed position using the trim lever, set the Spoiler control's **IDLETRIM** attribute to a suitable value.

#### *Disabling Spoiler via a Switch*

Sailplane pilots often use the left hand stick to control spoiler. However, if the same stick is used to operate the rudder, it's possible to nudge the spoiler accidentally, with undesirable consequences.

Consequently a switch is often used to disable the spoiler function during normal flight, enabling the spoiler only during the landing phase. There are two ways to program a switch to do this: the obvious way – and a better way.

Method 1 – disable mixer inputs

The obvious way is to use a switch say S1 to disable all "Spoiler" mixer inputs. To do this you do the following for each servo affected by the spoiler control:

1. Go to the **Servo Travel** screen
2. Go to the Spoiler page
3. Assign S1, so that when S1 is active, the spoiler input is disabled.
4. Repeat steps 1-3 for all servos affected by the spoiler control

There are two problems with this method:

First, we need to repeat this adjustment for several servos, in a six servo F3F model this can be very tiresome.

Secondly, this doesn't completely solve the problem if BUTTERFL mix is used. Because of a bug/feature in the operating system, aileron differential suppression (one of the special features of BUTTERFL) is affected by the spoiler stick position even if all spoiler inputs to the BUTTERFL mixer are disabled.

To demonstrate this:

1. Set up a model with the BUTTERFL mixer and with say 50% aileron differential
2. Disable the spoiler according to the instructions above.
3. Switch on the radio, and with full aileron applied, move the spoiler stick up and down…

See how the varying aileron differential suppression affects one of the ailerons, even though crow operation (where both ailerons go up) is suppressed correctly.

Method 2 – Using FIX VAL

A simpler solution is to use a secondary switch to enable Spoiler's **FIX.VAL.1** attribute. This also deals with the aileron differential suppression problem. Here's how it works:

With FIX VAL active, the spoiler control appears to be at a fixed position, *irrespective of the actual position of the stick*. By setting FIX VAL to the value corresponding to the "spoiler closed" position, we can effectively use the switch to disable the spoiler: in one position (FIX VAL enabled) the spoiler is closed, in the other (FIX VAL disabled) it's governed by the stick position as normal.

1. Go to the **Control Setup** screen
2. Choose the Spoiler control,
3. Choose the **FIX.VAL** attribute and set the value to zero, corresponding to the spoiler closed position.
4. Change the FIX VAL "mode" from OFF(inactive) to S1-5 (controlled by a switch).

Using this method, only one adjustment has to be made and aileron differential suppression is switched off when the spoiler is disabled as you would expect.

# Coupled Ailerons and Rudder

CAR is the traditional acronym for Coupled Ailerons and Rudder[9]. CAR mixing is always available if Rudder and Aileron controls have been assigned.

CAR is set in the **Combi Switch** screen. It operates in three modes – ON, OFF or controlled by a secondary switch. The default mode is OFF.[10]

Normally, it would be set up so that aileron is the primary control and rudder is the slave however it can be set for reverse operation.

CAR differs from user and built-in mixers as follows

- It has its own dedicated screen.
- It works with other mixers. For example, CAR can be added retrospectively to a set-up which uses a V.TAIL mixer.
- You do *not* need to take it into account when planning a new configuration – you can easily add it afterwards.

# The FIX VAL Virtual Control

The FIX VAL virtual control[11] is new in version 3.0 of the system software. The fly-sheet for v3.0 states "The FIXED VALUE facility also opens up some interesting new possibilities in conjunction with the USR-MIX mixers."

So what's the difference between the FIX VAL control, and the regular controls like Aileron, Elevator etc?

First, FIX VAL only provides a two-position function, it can't be used for proportional control.

Second, you don't assign it using the **Assign Control** screen. Instead you assign a servo to FIX VAL in the **Assign Servo** screen. Then you go to the **Servo Travel** screen and assign a secondary switch. When the switch is active, the servo moves to the position set in **Servo Travel**, when inactive, the servo goes to neutral as set in **Servo Centre**.

FIX VAL therefore provides a method of activating a servo for simple 2-position applications as an alternative to using a primary widget D-H. Note that whichever method you choose, you only get ½ servo travel (from neutral to one end-point). We'll see a way around this later on using a user mixer.

You can assign as many servos as you like to the FIX VAL control limited by the number of secondary switches and channels available. Each servo will see its own "clone" with its own centre and end point and switch mode.

Since FIX VAL is a control – albeit a strange one - it can be used as an input to a user mixer. And like other controls, you can even specify the FIX VAL control as *all* the inputs to the *same*

[9] CAR was common even before the days of computer radios, because as the name suggests, it's a coupling rather than a mixing function which can easily be implemented mechanically.

[10] It's acceptable to have CAR permanently on when if you are new to ailerons. However, for experienced pilots CAR hinders rather than helps co-ordinated turns. In a high aspect ratio sailplane for example, to do a turn without skid or slip, rudder is normally maintained throughout the turn, but ailerons are used only to roll into and out of the turn and neutralised or even reversed once the correct angle of bank has been achieved.

[11] Not to be confused with the **FIX.VAL** attribute, which is used for setting preset positions for some controls.

mixer. Each mixer input sees its own independent “clone” of FIX VAL. When you assign a servo, the **Servo Travel** shows a separate page for each input.

The use of FIX VAL in conjunction with user mixers does indeed provide some interesting possibilities including the ability to provide full servo travel from a standard 2-position switch – we’ll see an example later.

# Managing Model Memories

The 3030 provides to manage the 99 memories available. You can move from one model memory to another. You can erase memories. You can also copy one memory to another one. There are two main uses for this:

1. Provide a second or third configuration for the same model, for different phases of flight e.g. for F3B you might have launch, cruise, duration memories. By keeping these memories in consecutive memory locations, and assigning the models names with a number sequence in the last character, you can switch between memories using the M switch in flight. Bear in mind that if you need to re-centre a servo for any reason, you’ll need to repeat the adjustment across all the memories which is tiresome. Quite a lot can be accomplished by means of a single memory and changeover switches which is my preferred method for F3F. Your mileage may vary.
2. Use one memory as your “main” memory, and another as your “working” memory for experimenting with different settings etc. When you’re happy with your working memory, copy it to your main memory. This is my favoured approach.

The naming conventions in the manual are a little confusing, “File” is used in the program while “List” is used in the manual. They both refer to what is conventionally know as a model memory.

Memory management is one of the few weak areas in the system. Here are some gotchas.

First, Multiplex don’t provide a backup program to transfer configurations to a PC. This is essential in my view – I have heard of two users (of another brand radio) who lost all their settings when the Lithium backup battery failed. Another MPX owner almost ran over his set with his car. Fortunately there are a couple of third-party sources of backup software. mcTool from Airworld does this job very well. The software is simple and robust – their web site is http://www.airworld.de. At the time of writing the software is in German but they are considering an English version.

Neil Gillies of Seagull Technology produces sMPX which is a very reasonably priced shareware backup program for the 3030 with some neat features such as the ability to shuffle memories off-line. See http://www.sea-gull.demon.co.uk.

The second gotcha is a user-interface problem which makes it easy to accidentally erase memory 15. When you’re in the File Copy screen, if you cycle the copy mode past the “EXPORT” option and the current memory number is greater than 15, the memory number jumps to 15. If you don’t notice this, and delete what you think is your current model memory, you’ll end up deleting memory 15. The cure is not to use memory 15!

The third gotcha concerns copying a model memory in Full mode. The model name is copied along with the data, leading to duplicate names. When copying in Controls Only mode the original name is retained, so the user interface is rather inconsistent in this area.

That’s all I’ll say about model memories here - there’s a full treatment in the user manual.

# EXAMPLE 3: F3F machine

Let's now consider a sophisticated set-up – that of a racing F3F model. The purpose of this example is to demonstrate practical uses for some of the more advanced techniques, such as the secondary switches and Gx switch.

The example is based on my Ellipse 2V machine which is set up for F3F slope racing.

## The Requirements for F3F

Obviously different pilots have their own ideas about how a model should be set up. The first step for any complex model, or programming task for that matter, is to define the requirements. Here are my requirements for a V-tail model for F3F competition such as the Ellipse 2V.

**V-tail**

- Rudder/Elevator (V-tail) mixing.
- Coupled Ailerons and Rudder activated via switch or by flap setting.

**Wing**

- Crow brake on throttle stick, suppressible from a switch to prevent inadvertent operation during cruising flight.
- Elevator trim compensation for crow brake.
- 3 preset positions for flap: "speed", "normal", "thermal".
- Flaps applied to both inboard and outboard wing surfaces.
- Elevator → Flap ("snap flap") mixing. Automatically disabled when flaps are deployed.

## Control Assignments

The only change from the 60" Pylon Racer example is the addition of flaps which are activated by a 3-position switch on G.

**Table 13 – F3F control assignments**

| Control | Port (widget) |
|---|---|
| Rudder | A (left stick left/right) |
| Aileron | C (right stick left/right) |
| Elevator | D (right stick up/down) |
| Spoiler | B (left stick up/down) |
| Flap | G (3-position switch) |

## Servo Assignments

By this stage you should be familiar with the steps involved in building the servo assignment table. The method is exactly the same as the previous examples, there's just more of it! So I won't repeat the steps in detail.

**Table 14 – F3F servo assignments**

| Servo | Channel | Controls | Servo Assigned to | Servo Travel Pages |
|---|---|---|---|---|
| Aileron RIGHT | 1 | Aileron, Spoiler, Elevator,Flap | BUTTERFL | Aileron, Spoiler, Elevator(Gx), Flap |
| Aileron LEFT | 2 | Aileron, Spoiler, Elevator,Flap | BUTTERFL | Aileron, Spoiler, Elevator(Gx), Flap |
| Flap RIGHT | 3 | Spoiler, Elevator,Flap | BUTTERFL | Aileron(OFF), Spoiler, Elevator(Gx), Flap |
| Flap LEFT | 4 | Spoiler, Elevator,Flap | BUTTERFL | Aileron(OFF), Spoiler, Elevator(Gx), Flap |
| V-tail LEFT | 5 | Rudder, Elevator, Spoiler | V.TAIL+ | Rudder, Elevator, Spoiler, Flap(OFF) |
| V-tail RIGHT | 6 | Rudder, Elevator, Spoiler | V.TAIL+ | Rudder, Elevator, Spoiler, Flap(OFF) |

Channel numbers 1-4 are assigned alternately to Aileron Right/Left then Flap Right/Left. This is required for correct operation of aileron differential (see the user manual, p. 33).

There's a lot of mixing going on here. Let's see what's going on with the aileron servos, which have four entries in the Control column:

- Applying crow brake (**spoiler)** control lifts both ailerons.
- Applying up-**elevator** droops both the ailerons (snap-flap mixing).
- Applying **flaps** also droops the ailerons
- Applying **aileron** control moves the aileron servos (!).

Note the entry for "Elevator(Gx)" in the last column for the flap servos 5 and 6: Gx is used to suppress Snap Flap when flaps are in the thermal position. See below for details of Gx set-up.

## Programming Screens for F3F Example

The following programming screens are used in setting up F3F:

**Table 15 - programming screens used in F3F set-up**

| Program Screen | Notes |
|---|---|
| Assign Control | As in Table 13 |
| Assign Servo | As in Table 14 |
| Servo Travel | Set servo travel and mixing |
| Servo Centre | Set servo neutrals |
| Servo Limit | Limit movement of V-tail servos when both Rudder and Elevator are applied together.<br><br>Limit movement of Up-going aileron when both Spoiler (crow) and Aileron are applied. |
| Control Setup | See below for control attributes. |

**Table 15 (cont.)**

| | |
|---|---|
| Combi Switch | CAR mode = ON or controlled by secondary switch.<br><br>One possible variation would be to control CAR via Gx, so that CAR is disabled when flaps are in thermal position[12]. |
| Soft Switch | The Gx "virtual" secondary switch is slaved off the flap widget "G". When flaps are in thermal position, Gx is OFF, otherwise Gx is ON.<br><br>In the Servo Travel screen, Gx is used to control Elevator→Flap mixing. |

## Control Attribute Settings for F3F Example

**Table 16 – Control attributes for F3F set-up**

| Control | Task | Attribute | VALUE |
|---|---|---|---|
| Aileron | Set differential. | DIFFER | 50 |
| Spoiler | See "Disabling Spoiler via a Switch" (page 21) | FIX.VAL.1 | 0 |
| | | NORMPOS | ↑ |

# More About User Mixers

Let's now take a closer look at user mixers.

As we have seen, built-in mixers like ELEV.MIX, BUTTERFL etc. are fine for many applications. However flyers of scale and aerobatic models often have more complex requirements which can only be satisfied with custom mixers, i.e. mixers with a specific set of inputs for the particular application. In the following section, we'll see how to design and set up user mixers.

To understand this section, you will need to have read What are Mixers? (page 7), if not please refer back to it first.

## Setting up a User Mixer

There are three user mixers designated USR-MIXn where n is 1, 2 or 3. A user mixer can have up to four control inputs. Any of the standard controls like Aileron and Flap can be used, as well as the Fix Val virtual control. You can even assign the same control to more than one input.

To create or modify a user mixer, you use the **UserMix** screen. This is where you specify inputs and trim mode.

[12] This would allow more precise thermalling during the F3F climbout phase, while providing coupling for snappier entry into turns during the racing legs.

Note that the mixer is simply a list of control inputs, we say nothing about servos in the **UserMix** screen - mixers are associated with servos in the **Assign Servo** screen. Another point important worth mentioning here: when a user mixer is created or modified, its definition is available to *all* model memories not just the one it was set up in.[13] For more details of the **UserMix** screen, see *Appendix H - Program Screen*.

## Activating a User Mixer

You activate a mixer by assigning a servo to it. When you assign a servo, the system looks at the mixer definition, and sets up "pages" in the **Servo Travel** screen, one page for each mixer input. This occurs at the moment you press the **M** key in the **Assign Servo** screen.

## Recycling

There are only three user mixers, so does that mean you are restricted to three possible mixing schemes for all your servos? Fortunately not, the reason follows from the previous section.

Say you create a user mixer with Elevator and Rudder inputs, assign servo 1 to it, then add Spoiler to the list of inputs. Remember that servo 1 will not "see" a page for the new Spoiler input until you re-assign the servo to the user mixer.

What if instead of reassigning the first servo, you assign servo 2 to the mixer? This leaves servo 1 with the original (before the changes) mixer inputs, and servo 2 with the new combination of inputs including Spoiler. Using this technique one mixer can provide a different mixing scheme for two, three or even all your servos. I call this technique "re-cycling".

Just to recap,

1. After you create or alter a user mixer, you must assign a servo to it to make it effective. To reinforce the point, the system takes you straight to the **Assign Servo** screen when you hit the **M** key in the **UserMix** screen.

2. You can "recycle" a user mixer by changing the inputs and assigning it to a *different* servo.

[13] It's even available in supposedly "—EMPTY--" model memories. You can demonstrate this as follows:

1. Make a copy of one of your model memories, using the File Copy screen, and call it say TEST.
2. Go to the UserMix screen and define a user mixer say USR-MIX1 with say Elevator and Aileron inputs.
3. Delete the TEST model memory (using the FileCopy screen). Note that the model name is automatically changed to "Empty".
4. Go into the UserMix screen, and take a look at USR-MIX1. It still has the same Elevator and Aileron inputs, i.e. it is still there, even though we deleted the model memory!

So a user mixer can be used in two ways.

- As a “permanent” mixer. If you have a combination of inputs which you would like to use for several models, then reserve a user mixer for this purpose. User mixers are available across all model memories, so you can use it exactly as you would a predefined mixer.
- As a “recycleable” mixer. If you have lots of complex models with more than three mixing schemes in total, then re-cycle one or more user mixers as described above.

## Example 1 – “Knife Edge Plus” mixer

Our first example is a sophisticated mixer for an aerobatic model[14]. The requirements are:

- Knife-edge facility: rudder➔elevator and rudder➔aileron mixing controlled via S5.
- Engine side/down thrust compensation: Throttle➔elevator and throttle➔rudder are permanently mixed to simulate engine thrust line adjustment.
- Loop tracking: Elevator➔aileron mixing is also permanently enabled.

Lets build the servo assignment table. We will keep an open mind at this stage as to whether we really need a user mixer to do the job.

Considering the requirements above, we do our trick of waggling our imaginary sticks and seeing which servos respond, and end up with the following table:

**Table 17 - Servo Assignment Table with Controls**

| Servo | Controls | Servo assigned to | Servo travel pages |
|---|---|---|---|
| Aileron | Aileron<br>Elevator<br>Rudder | | |
| Elevator | Throttle<br>Elevator<br>Rudder | | |
| Rudder | Throttle<br>Rudder | | |
| Throttle | Throttle | | |

Looking at column 2, we see that mixers must be used for all controls except Throttle. We will ignore the Throttle for the rest of the discussion as it does not involve a mixer.

Next look at *Appendix G - Predefined Mixers* to confirm there are no suitable pre-defined mixers.

The next step is a new one, specific to user mixers: we need to decide on the trim mode for each input. If you want to include the trim lever in the mixer input signal, then the trim mode must be set to ON. Normally you would only want to set the trim mode to On for the *primary* input for a particular servo e.g. the Elevator input for the elevator servo. Note that if you need the trim mode to be on for some servos and off for others, you will need two separate inputs e.g. Elevator and Elevator+T.

[14] With thanks to Harry Curzon

**Table 18 - Servo Assignment Table with Trim Mode**

| Servo | Controls | Servo assigned to | Servo travel pages |
|---|---|---|---|
| Aileron | Aileron+T<br>Elevator<br>Rudder | | |
| Elevator | Throttle<br>Elevator+T<br>Rudder | | |
| Rudder | Throttle<br>Rudder+T | | |

Now gather up the controls in column 2.

- Aileron
- Aileron+T
- Elevator
- Elevator+T
- Rudder
- Rudder+T
- Throttle.

These will be our mixer inputs. Since there are more than four, we will have to split them up between two or more user mixers since a single user mixer can only have a maximum of four inputs. The obvious solution is to have one user mixer per servo.

**Table 19 - Servo Assignment Table with Three User Mixers**

| Servo | Controls | Servo assigned to | Servo travel pages |
|---|---|---|---|
| Aileron | Aileron+T<br>Elevator<br>Rudder | **User-Mix 1, inputs:**<br>**Aileron+T**<br>**Elevator**<br>**Rudder** | **Aileron ON**<br>**Elevator ON**<br>**Rudder S5** |
| Elevator | Throttle<br>Elevator+T<br>Rudder | **User-Mix 2, inputs:**<br>**Throttle**<br>**Elevator+T**<br>**Rudder** | **Throttle ON**<br>**Elevator ON**<br>**Rudder S5** |
| Rudder | Throttle<br>Rudder+T | **User-Mix 3, inputs:**<br>**Throttle**<br>**Rudder+T** | **Throttle ON**<br>**Rudder ON** |

You'll see that we are using all three user defined mixers, one for each servo. However, there is an alternative: we could use a single mixer and recycle it by performing the following sequence.

1. Create USR-MIX1 with Aileron+T, Elevator and Rudder inputs.
2. Assign Aileron servo to USR-MIX1
3. Edit USR-MIX1 so it has Throttle, Elevator+T and Rudder inputs.
4. Assign Elevator servo to USR-MIX1.
5. Edit USR-MIX1 so it has Throttle, and Rudder+T inputs.
6. Assign Rudder servo to USR-MIX1.

**Table 20 – Servo Assignment Table Using Recycled USR-MIX-1**

| Servo | Controls | Servo assigned to | Servo travel pages |
|---|---|---|---|
| Aileron | Aileron+T<br>Elevator<br>Rudder | User-Mix 1, inputs:<br>Aileron+T<br>Elevator<br>Rudder | Aileron ON<br>Elevator ON<br>Rudder S5 |
| Elevator | Throttle<br>Elevator+T<br>Rudder | User-Mix 1, inputs:<br>Throttle<br>Elevator+T<br>Rudder | Throttle ON<br>Elevator ON<br>Rudder S5 |
| Rudder | Throttle<br>Rudder+T | User-Mix 1, inputs:<br>Throttle<br>Rudder+T | Throttle ON<br>Rudder ON |

We have now effected all the mixing using just one user mixer definition. This is fine for a one-off setup but not very convenient if we have several models using the same mixing scheme - we'd have to go through the whole process of editing USR-MIX1 before assigning each servo.

This inconvenience can be overcome if we are prepared to compromise a little. As you saw earlier, Aileron and Aileron+T are two different inputs as far as a user mixer is concerned. What if we made *all* the controls "+T"? We would then have just four inputs: Aileron+T, Throttle+T, Elevator+T and Rudder+T. Since the maximum number of inputs for a user mixer is four, we could then use a single mixer for all servos *without recycling* as follows:

**Table 21 - Servo Assignment Table with "+T" on All Controls**

| Servo | Controls | Servo assigned to | Servo travel pages |
|---|---|---|---|
| Aileron | Aileron+T<br>Elevator+T<br>Rudder+T | **User-Mix 1, inputs:<br>Aileron+T<br>Elevator+T<br>Rudder+T<br>Throttle+T** | **Aileron ON<br>Elevator ON<br>Rudder S5<br>Throttle OFF** |
| Elevator | Throttle+T<br>Elevator+T<br>Rudder+T | **As above** | **Aileron OFF<br>Elevator ON<br>Rudder S5<br>Throttle ON** |
| Rudder | Throttle+T<br>Rudder+T | **As above** | **Aileron OFF<br>Elevator OFF<br>Rudder ON<br>Throttle ON** |

Note that inputs which are Off in the last column will have to disabled explicitly in the **Servo Travel** screen.

We have implemented the whole scheme using a single user mixer. Beware though that we have compromised by enabling the trim mode "+T" for all inputs. This means that trim movement on all the controls will be included in *all* the outputs, e.g. moving the elevator trim will cause a displacement of the aileron. Often as in this example it will be a very small second order effect - whether this is acceptable depends on the particular application.

## Mimicking Built-in Mixers

You can cook your own version of the BUTTERFL mixer - complete with Aileron Differential Suppression[15] - by setting up a user mixer with Aileron+T, Elevator, Flap+T and Spoiler inputs. This could be useful if you want to create your own variations on the built-in mixer.

# Advanced Techniques

This section puts together some of the features of the system to do some tricks. You may not need to use these functions in your models, but if you understand how they work you can devise your own solutions.

## Switching Mixer Inputs

A secondary switch can be used to activate either of two mixer inputs. This is a very powerful technique with a number of applications. The best way to illustrate this is by an example.

***Example – Idle Up***

On some large power models, it is required to have two different ranges of throttle travel, one for "flight idle" and the other for "ground idle", selectable via a switch. Typically the top end is the same for both settings of the switch, only the low range is different. This is how you can do it with S1:

1. Go to the **USER MIX** screen and set up USR-MIX1 with two identical inputs: Input 1 = Throttle + T, Input 2 = Throttle + T. Note that T must be specified for Throttle's IDLE TRIM attribute to work via the trims.

2. Assign the throttle servo to USR-MIX1

3. The **Servo Travel** screen for the throttle servo will have two pages, both of them for Throttle (since we have specified Throttle for *both* inputs to the user mixer). Go to the first page, and set the mode to S1↑. With S1 in the up position, adjust the servo end points for full throttle and *ground* idle.

4. Go to the second Throttle page. Set the mode to S1↓. With S1 in the down position, adjust one end point for full throttle as above, and the other end for *flight* idle.

And that's it! With S1 up, the first input (ground idle) is selected, the second (in-flight idle) is disabled. Vice versa with S1 down. Note that the IDLE TRIM attribute functions through the mixer.

**Table 22 – Servo assignment table for Idle Up mixer**

| Servo | Controls | Servo assigned to | Servo Travel Pages |
|---|---|---|---|
| Throttle | Throttle | User-Mix 1, inputs:<br>Throttle + T<br>Throttle + T | Throttle S1 ↑<br>Throttle S1 ↓ |

[15] Aileron Differential Suppression is touted as a "feature" of the BUTTERFL mixer. In fact, ADS is active with any user mixer which has both aileron and spoiler inputs.

## Using SI to Switch or Disable Mixer Inputs

In the previous example, we saw how to switch between mixer inputs, using a two position switch. In this example we see how to switch between two inputs, and additionally disable them both. For this a three position switch is required in conjunction with SI (that's SI as in "ess-eye") – note this use of SI is not documented in the user manual.

***Example: +ve and –ve snapflap switch.***[16]

For advanced aerobatics on a scale power machine, it is required to have a three position switch controlling Elevator→Flap mixing as follows:

- UP: Elevator to Flap mixing, surfaces move in *opposite* directions
- DOWN: Elevator to Flap mixing, surfaces move in *same* direction
- MIDDLE: Elevator to Flap mixing is disabled.

Here's how it's done:

1. Create a user mixer say USR-MIX1, with three inputs: Flap, Elevator and Elevator (Elevator is specified twice).

2. Assign one or more flap servos to USR-MIX1

3. Go to the Servo Travel screen for one of the flap servos. Set the mix:

- Flap input: set the required movement and set the mode to ON.
- First Elevator input: set the required Elevator→Flap mixing and set the mode to SI↑.
- Second Elevator input: set the required Elevator→Flap mixing set the mode to SI↓.

4. Repeat 3 and 4 for the other flap servo.

With SI in the Up position, Flap and the first Elevator input are enabled. With SI Down, Flap and the second Elevator input are enabled. With SI in the middle (OFF), both Elevator inputs are *dis*abled and the flaps are controlled purely from the Flap input.

## Using SI with Flaps

Here's a way of getting four preset positions for flaps. This could be useful for any applications, where four different flap settings are required for e.g. launch, duration, cruise and speed. The method involves a two position switch for the main Flap control, with a three position switch on SI providing two further preset positions.

| **SI (3-pos switch)** | **Flap control (2-pos switch)** | **Flap Position** |
|---|---|---|
| MIDDLE | Up | Determined by Servo Travel 1st end-point |
| MIDDLE | Down | Determined by Servo Travel Centre |
| UP | | Determined by Flap's FIX.VAL.1 attribute |
| DOWN | | Determined by Flap's FIX.VAL.2 attribute |

[16] Thanks to Harry Curzon for posing the problem *and* the solution.

## How to Get Full Travel Using a 2-position Switch.

It is not possible to obtain full servo movement from a simple switched function using a standard 2-position two-wire switch as supplied with the set.

There are two ways of getting round this limitation

Method 1 - using the FIX VAL control, and a user mixer.

1. Go to the **User Mixer** screen
2. Choose the first mixer "USR1" and assign the FIX VAL control to inputs 1 and 2.
3. Go to the **Assign Servo** screen and assign servo 1 to USR1.
4. Go to the **Servo Travel** screen. There should be two FIX VAL pages, which will look identical so you may not notice that you are stepping from one to the other as you switch pages. Just choose the first one at this stage.
5. Set the travel to –100% and assign the switch S1 to it so it looks like this: **S1↓**
6. Go to the other FIX VAL page.
7. Set the travel to +100% and assign switch S1 so it looks like this: **S1↑**

S1 should now move the servo between its end points. Note the servo centre in the **Servo Centre** screen should be set to zero.

The way it works is this:

- The mixer has two inputs, each seeing an independent copy of FIX VAL.
- With S1 down, one mixer input sees the FIX VAL of –100, the other input is disabled (0). Add the inputs and we get -100 at the output.
- With S1 up, the first input is disabled, while the other input sees the second FIX VAL(100). Add them together and you get +100 at the output.
- Operating switch S1 therefore alternates the outputs between –100 and +100 i.e. full travel either side of neutral.

Note that this method doesn't use up any valuable controls like AUX or FLAP, it works purely via a single secondary switch.

Method 2 – using a three wire harness

The second method doesn't involved any special programming. It requires a third wire to be added to a standard two-position SPDT switch (as supplied for the secondary switches). Solder the wire between the spare tag on the switch and the spare tag on the port plug.

To test it, plug the harness into port H, assign AUX to the H, assign a servo to AUX. Toggle the switch, and the servo should now have full travel either side of neutral. Note that you will not be able to use the modified switch for secondary switch applications after this modification (although no damage will be done if you try).

## A Binary Servo Controller

This is a solution looking for a problem, but it does illustrate what can be done with a little thought. The idea is to enable switches S1 to S4 to move a servo to any one of 16 positions as follows:

- All switches off = binary 0000 ➔ servo is at centre
- All switches on = binary 1111 = $15_{10}$ ➔ servo is at one end point.
- Other configurations ➔ servo position is determined by value 0 – 15.

The solution is as follows:

1. Create USR MIX 1 with four inputs, all of them FIX VAL control.
2. Assign Servo 1 to USR MIX 1 in the Assign Servo screen.
3. Go to the Servo Travel + Rev screen and select Servo 1. There will be four "pages", one for each copy of FIX VAL input. Initially they will all be identical so there will only appear to be one page! Plough on nevertheless...
4. For page 1, set travel to 48 and assign to S1
5. For page 2, set travel to 24 and assign to S2
6. For page 3, set travel to 12 and assign to S3
7. For page 4, set travel to 6 and assign to S4

# DIY Switches

In this section we'll investigate how to wire up DIY switches for the various ports. We'll divide the ports into three groups namely E-I[17] (primary switches), S1-S5 (two-state secondary switches) and SI (three-state secondary switch).

## Switches

All switches on the mc3030 are SPDT type. Both 2-way and 3-way switches are used. Switches may be mechanically biased to one side via an internal spring, but this does not affect the electrical properties.

## Harnesses

Switch harnesses may be 2-wire or 3-wire.

## Port Logical States

Ports S1 – S5 have two states 0 and 1.

Ports E-H and SI have three states 0, 1 and 2.

Note the numbers are only used to indicate the possible states, the actual values are not significant. The state of a port depends on the state of the switch to which it is connected, which in turn depends on the position of the switch and the number of wires (2 or 3) in the harness.

[17] A thru D as are reserved for the stick units.

**Table 23 – Port States**

| Switch type | Switch Position | State of S1-S5 | State of E-H, and SI |
|---|---|---|---|
| 2-way 2-wire | Up | 1 | 1 |
| | Down | 0 | 0 |
| 2-way 3-wire | Up | 1 | 1 |
| | Down | 1 | 2 |
| 3-way 2-wire | Up | 1 | 1 |
| | Middle | 0 | 0 |
| | Down | 0 | 0 |
| 3-way 3-wire | Up | 1 | 1 |
| | Middle | 0 | 0 |
| | Down | 1 | 2 |

## Port states and system software

The following tables show the effect of the different port states.

**Table 24 - Servo assigned to AUX, AUX assigned to 2 position switch on E-H.**

| Port State | Servo Position |
|---|---|
| 1 | End point 1 |
| 2 | End point 2 |
| 0 | Neutral |

**Table 25 – Logic for S1–5 & L/S**

| Port State | Software state | |
|---|---|---|
| | Normal | Rev |
| 1 | ↑ | ↓ |
| 0 | ↓ | ↑ |

**Table 26 - Logic for port SI**

| Port State | Software state | |
|---|---|---|
| | Normal | Reverse |
| 1 | ↑ | ↓ |
| 2 | ↓ | ↑ |
| 0 | Neither | Neither |

## Making Your Own Plug/Switch Assemblies

My mc3030 was supplied with gold-plated switches by manufactured by C&K. These are available from the Farnell catalogue.

**Table 27 - Farnell part numbers for SPDT sub-miniature switches**

| Action | Positions | Farnell Part No. |
|---|---|---|
| On/On | 2 | 151-165 |
| On/Off/On | 3 | 151-167 |
| On/Mom | 2 | 917-930 |

For information on making up the plugs which connect to the mainboard, I'm indebted to Harry Curzon: "The part I used from Maplins to make plugs for switches is a 'straight bipolarised locking plug assembly' at 0.1inch pitch, meant for pc boards. It is a strip of plastic with pins pushed through, solder the lead on to one end and the other goes into the sockets in the mc3030 mainboard. For a 3-pin assembly the part is BX96E at 45p. Much better value is the 12 pin assembly YW14Q at 70p which can be carefully cut into 4 plugs."

# Appendix A - Mainboard Connections

## Ports for Primary Sticks & Switches

| PCB Socket | Code | Type of Widget Allowed | Default Factory Connection |
|---|---|---|---|
| KnR[18] | A,B | 2-axis stick with trims | Stick without ratchet<br>A=←→ B= ↑↓ |
| KnL | C,D | 2-axis stick with trims | Stick with ratchet<br>C=←→ D=↑↓ |
| E | E | Slider or 2/3 position switch | Slider |
| F | F | Slider or 2/3 position switch | Slider |
| G | G | Slider or 2/3 position switch | 3-position switch |
| H | H | Slider or 2/3 position switch | (no widget supplied) |
| I | I | Slider or 2/3 position switch | (no widget supplied) |

## Secondary Switches

| PCB Socket | Switch Code | Type of Widget Allowed | Default Factory Connection |
|---|---|---|---|
| S1 | S1 | 2 position switch | 2 position |
| S2 | S2 | 2 position switch | 2 position |
| S3 | S3 | 2 position switch | 2 position |
| S4 | S4 | 2 position switch | (no widget supplied) |
| S5 | S5 | 2 position switch | 2 position |
| I | SI | 2 or 3 position switch | (no widget supplied) |

## Miscellaneous Ports

| PCB Socket | Application |
|---|---|
| M | 3-position switch for switching model memories. |
| DE | Digi Adjuster |
| LS | Teacher/pupil (“buddy box”) switch. |
| MNT | Multinaut module |

[18] The dual-axis sticks each have identical 5-pin plugs which can apparently be swapped round, however this is not a good idea. If you need to switch from Mode 1 to Mode 2 (throttle right to throttle left) then the ratchets should be swapped as described in the manual.

# Appendix B - Controls and Attributes

This table shows all the controls selectable **Assign Control** screen, and their attributes.

| **Controls** | | **Attributes** | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | Applic. | EXPO | DUAL RATE | TRAVEL | TRAVEL +/- | CENTRE | DIFFER | FIX. VAL. 1 | FIX. VAL. 2 | NORM POS | IDLE-TRIM | THR. CURVE. | DIR. THRO. | SUPP-RESS |
| Aileron | | Y | Y | Y | | Y | Y | | | | | | | |
| Elevator | | Y | Y | | Y | Y | | | | | | | | |
| Rudder | | Y | Y | | Y | Y | | | | | | | | |
| Throttle | | | | | | | | Y | | | Y | | | |
| Throtl 2 | | | | | | | | | | | Y | | | |
| Spoiler | | | | | | | | Y | | Y | Y | | | |
| Flap | | | | | Y | | | Y | Y | Y | | | | |
| Retract | | | | | | Y | | Y | | | | | | |
| Towhook | | | | | | Y | | Y | | | | | | |
| Mixture | Heli | | | | | Y | | | | | | | | |
| Roll | Heli | Y | Y | | Y | Y | | | | | | | | |
| Nose ↑/↓ | Heli | Y | Y | | Y | Y | | | | | | | | |
| Yaw | Heli | Y | Y | | Y | Y | | | | | | | | |
| Pitch | Heli | Y | | | Y | Y | | | | | | Y | Y | |
| Gyro | Heli | | | | Y | Y | | Y | | | | | | Y |
| Ship Rudd | Boat | Y | Y | | | Y | | | | | | | | |
| Motor | E-flite | | | | | Y | | | | | | | | |
| Motor2 | E-flite | | | | | Y | | | | | | | | |
| Aux.1 | | | | | Y | Y | | Y | | | | | | |
| Aux.2 | | | | | Y | Y | | Y | | | | | | |
| Aux.3 | | | | | Y | Y | | Y | | | | | | |
| Aux.4 | | | | | Y | Y | | Y | | | | | | |

# Appendix C - Screens Navigation

| Screen Name | Navigation |
|---|---|
| Servo Travel | Servo→Trvl+Rev |
| Servo Centre | Servo→Centre |
| Servo Limit | Servo→Limit |
| Servo Test | Servo→Test |
| Control Setup | Control→Setup |
| Control Slow | Control→Slow |
| Combi Switch | Control→Combi-sw. |
| Test Widgets | Control→Test |
| File Copy | Files→Copy |
| File Name | Files→Name |
| File Shift | Files→Shift |
| Check Trims | Files→Chktrim |
| Timer | Menu2→Timer |
| Op Time | Menu2→OP.time |
| Assign Control | Menu2→Assign→Control |
| Assign Servo | Menu2→Assign→Servo |
| UserMix | Menu2→Assign→UserMix |
| Soft Switch | Menu2→Assign→Softsw. |
| Pupil | Menu2→Menu3→Pupil |
| Teacher | Menu2→Menu3→Teach |
| RPM | Menu2→Menu3→RPM |
| Transmission Mode | Menu2→Menu3→PCM/PPM |

# Appendix D - Program Screens by Category

| Category | Screen | Navigation |
|---|---|---|
| Assigning Controls | **Assign Control** | Menu2→Assign→Control |
| Assigning Servos | **Assign Servo** | Menu2→Assign→Servo |
| Servo Adjustments | **Servo Travel** | Servo→Trvl+Rev |
| | **Servo Centre** | Servo→Centre |
| | **Servo Limit** | Servo→Limit |
| Control Attributes | **Control Setup** | Control→Setup |
| Mixers | **User Mix** | Menu2→Assign→UserMix |
| | **Combi Switch** | Control→Combi-sw. |
| Memory Management | **File Copy** | Files→Copy |
| | **File Name** | Files→Name |
| | **File Shift** | Files→Shift |
| Timers | **Timer** | Menu2→Timer |
| | **Op Time** | Menu2→OP.time |
| Gx | **Soft Switch** | Menu2→Assign→SoftSwitch |
| Miscellaneous | **Servo Test** | Servo→Test |
| | **Control Slow** | Control→Slow |
| | **Test Widgets** | Control→Test |
| | **Check Trims** | Files→Chktrim |
| | **Pupil** | Menu2→Menu3→Pupil |
| | **Teacher** | Menu2→Menu3→Teach |
| | **RPM** | Menu2→Menu3→RPM |
| | **Transmission Mode** | Menu2→Menu3→PCM/PPM |

# Appendix E - Servo Assignment Targets

The items which servos can be assigned to, in the **Assign Servos** screen.

| Item | Type |
|---|---|
| AILERON | Control |
| ELEVATOR | Control |
| RUDDER | Control |
| THROTTLE | Control |
| THROTT 2 | Control |
| SPOILER | Control |
| FLAP | Control |
| RETRACT | Control |
| TOWHOOK | Control |
| MIXTURE | Control |
| ROLL | Control |
| NOSE ↑/↓ | Control |
| YAW | Control |
| PITCH | Control |
| GYRO | Control |
| SHIP RUDD | Control |
| MOTOR | Control |
| MOTOR2 | Control |
| AUX.1 | Control |
| AUX.2 | Control |
| AUX.3 | Control |
| AUX.4 | Control |
| FIX.VAL | Virtual Control |
| --------- | Undocumented, sends servo to one end of travel. |

| Item | Type |
|---|---|
| ELEV. MIX | Mixer |
| V. TAIL | Mixer |
| V.TAIL+ | Mixer |
| FLAPERON | Mixer |
| BUTTERFL | Mixer |
| SNAPFLAP | Mixer |
| QUADRO | Mixer |
| DELTA | Mixer |
| TAILROT. | Mixer |
| HEIM-MIX | Mixer |
| FLAREMIX | Mixer |
| HEAD-MIX | Mixer |
| DYN.THR. | Mixer |
| USR-MIX1 | User Mixer |
| USR-MIX2 | User Mixer |
| USR-MIX3 | User Mixer |
| UNUSED | (channel output = centre) |

# Appendix F - Secondary Switch Functions

The table shows which control attributes are controllable via secondary widgets.

| Attribute | Can be enabled/disabled by secondary switch? |
|---|---|
| EXPO | No |
| DUAL RATE | Yes |
| TRAVEL | No |
| TRAVEL +/- | No |
| CENTRE | No |
| DIFFER | No |
| FIX. VAL. 1 | Yes |
| FIX. VAL. 2 | Only using SI |
| NORM POS | No |
| IDLE-TRIM | No |
| THR. CURVE. | No |
| DIR. THRO. | No |
| SUPP-RESS | No |

# Appendix G - Predefined Mixers

| Mixer | Control Inputs | Recommended Servo Assignment | Application |
|---|---|---|---|
| ELEV MIX[19] | Elevator<br>Spoiler<br>Flap | Elevator servo | Conventional tail models with elevator trim compensation for spoiler/flaps. |
| V.TAIL | Elevator<br>Rudder | V-tail servos | Simple V-tail models |
| V.TAIL + | Elevator<br>Rudder<br>Spoiler<br>Flap | V-tail servos | V-tail models with elevator trim compensation for spoiler/flaps. |
| FLAPERON | Aileron<br>Flap | Wing servos | Combined flaps/ailerons |
| BUTTERFL[20] | Aileron<br>Flap<br>Spoiler<br>Elevator | Two or four servos in the wing. | Models with four wing servos and crow brakes<br><br>Models with two wing servos and aileron brakes<br><br>Also provides Snap Flap mixing (elevator→flap + elevator→aileron) and<br><br>Use in conjunction with ELEV MIX or V.TAIL+. |
| QUADRO | Aileron<br>Flap<br>Elevator | Wing Servos | Separate flaps and ailerons, with optional snap flap mixing (elevator→flap). Crow brakes not supported. |
| DELTA | Aileron<br>Elevator | Elevon servos | Delta and tailless models. |
| TAILROT. | | | HELI |
| HEIM-MIX | | | HELI |
| FLAREMIX | | | HELI |
| HEAD-MIX | | | HELI |
| DYN.THR. | | | HELI |

[19] Wrongly labelled ELEVATOR+ on p. 57 of the manual

[20] Wrongly labelled “aileron brake (Crow) mixer” on p. 58 of the manual

# Appendix H - Program Screens

| Screen Name | Navigation |
|---|---|
| **Servo Travel** | **Servo→Trvl+Rev** |

Description

Adjusts the travel and direction of a servo for each mixer input.

Operation

| | | |
|---|---|---|
| ◤ | +/- | Cycles over servos 1 – 9. The display shows the control or mixer the servo is assigned to. If the servo is not assigned to a control or mixer UNUSED is displayed. |
| ◥ | +/- | If the servo is assigned to a control pressing +/- has no effect.<br><br>If the servo is assigned to a mixer, +/- cycles over mixer inputs. |
| ◣ | +/- | Adjusts the amount of servo travel. The display shows the following fields:<br><br>A number between –110 and +110 representing the amount of travel. The sign indicates the direction of rotation of the servo.<br><br>Code A-I of widget associated with the control.<br><br>An arrow indicating which side of neutral this adjustment applies.<br><br>To adjust servo travel for the other side of neutral:<br>1. Move the widget until the arrow changes direction,<br>2. Adjust travel as above.<br><br>Note: if “CONTROL?” is displayed in this area, it means that the control has not been assigned to a stick/switch. This is not necessarily a problem, it just means that the page is disabled. |
| | R | Reverses rotation of servo for *both sides* of stick/switch travel. To reverse direction for one side only, use the +/- keys as above. |
| ◢ | +/- | Cycles through modes: ON, OFF and changeover switch.<br>• Mode = ON: control or mixer input is active.<br>• Mode = OFF: control or mixer input is ignored.<br>• Mode = SECONDARY SWITCH: control or mixer input is enabled by a switch. |
| | R | If mode is OFF or ON, flips between modes. If in “secondary switch” mode, reverses the sense of the switch. |

Notes

The Mode adjustment allows mixing to be switched on, off or controlled by a switch.

The Servo Travel adjustment can also be used to make the MPX compatible with other makes, so that switching transmitters between makes does not cause significant difference in servo travel. For Futaba/JR/Hitec/Sanwa compatibility, start off with a base Servo Travel value of around 75% each way. See also Servo Centre.

Program Screens (cont)

| Screen Name | Navigation |
|---|---|
| **Servo Centre** | **Servo→Centre** |

Description

Adjust the centre position of a single servo. Use this adjustment for fine tuning the position of control surfaces on the bench, before flight, or to compensate for bent or knocked linkages on the field.

Operation

| | | |
|---|---|---|
| ◤ | +/- | Cycles through servos 1-9. The display shows the control or mixer the servo is assigned to. If the servo is not assigned to a control or mixer UNUSED is displayed. |
| ◥ | +/- | Adjust the centre position between 0-110. |

Notes

To maintain the full range of servo movement, keep the centre value in the range –10 to +10. Between 10 and 110, the amount of servo travel on one side decreases linearly to zero (it is not changed on the other).

A centre value of –10 corresponds to a pulse width of 1.5ms, which is the standard for JR/Hitec/Futaba. Setting the centre to –10 for all servos allows MPX or other transmitters to be used without re-splining the servos.

When ◥ is pressed, any mixing and trim values for the servo are temporarily cancelled to move the servo to its true centre. As a result, the servo may appear to jump on entering and again on leaving the Servo Centre. This is normal behaviour.

This screen is not suited for trim lever adjustments, for that use adjust the Centre attribute.

| Screen Name | Navigation |
|---|---|
| **Servo Limit** | **Servo→Limit** |

Description

Places an absolute limit on the displacement of a servo.

Operation

| | | |
|---|---|---|
| ◤ | +/- | Cycles through servos 1-9. The display shows the control or mixer the servo is assigned to. If the servo is not assigned to a control or mixer UNUSED is displayed. |
| ◢ | +/- | Sets the limit between 0 – 110. The number represents a displacement from the neutral position.<br><br>The limit can be set for each direction of travel independently. Set the limit for one side, then move the associated stick/switch in the other direction until the arrow changes and set the other side. |

Notes

Limit is used most commonly in conjunction with mixers where applying maximum inputs simultaneously might cause control hinges to bind or other unwanted effects.

Changing the servo centre will necessitate a readjustment of the Limit value.

## Program Screens (cont)

| Screen Name | Navigation |
|---|---|
| **Servo Test** | **Servo→Test** |
| Description | |
| Exercises the servos assigned to a control. This screen is useful for checking servo travel without having to handle the transmitter. | |
| Operation | |
| +/- | Cycles between controls/mixers |
| R | Sets test mode on or off |
| Notes | |
| When in test mode, the MPX operating system takes over the selected control - operating the stick/knob manually has no effect.<br><br>If the selected control is an input to a mixer, *all* the servos assigned to the mixer are exercised. | |

| Screen Name | Navigation |
|---|---|
| **Control Setup** | **Control→Setup** |
| Description | |
| Adjust control attributes. There is a separate page for each attribute. | |
| Operation | |
| +/- | Cycles between sticks/switches A-H. The name of the associated control is displayed alongside. If no control has been assigned to the stick/switch, “UNUSED” is displayed. |
| +/- | Cycles between the attributes of selected control. |
| Notes | |
| See *mc3030 User Manual (pp. 37-43, 64-67)*. | |

Program Screens (cont)

| Screen Name | Navigation |
|---|---|
| **SoftSwitch** | **Menu2→Assign→Servo** |
| Description | |
| Assign a servo to a control | |
| Operation | |
| +/- | Cycles through servos 1-9. The display shows the control or mixer the servo is assigned to. If the servo is not assigned to a control or mixer UNUSED is displayed. |
| +/- | Changes the control or mixer the servo is assigned to. |
| Notes | |
| A servo cannot be assigned to more than one control or mixer. However… Several servos can be assigned to the same control or mixer (equivalent to using a Y-lead). | |
| See also | |
| *Appendix E - Servo Assignment* Targets | |

| Screen Name | Navigation |
|---|---|
| **Control Slow** | **Control→Slow** |
| Description | |
| Sets the servo speed to between 0.4 and 10 secs in each direction. | |
| Operation | |
| +/- | Cycles between controls. Both the widget code A-I and control name are displayed. |
| +/- | Adjust the servo transit time. Can be adjusted for each direction independently. |
| Notes | |
| Applies to controls assigned to any widget except A and C. | |
| To disable servo slow set the transit time to zero. | |

Program Screens (cont)

| Screen Name | Navigation |
|---|---|
| **Combi Switch** | **Control→Combi-sw.** |
| Description | |
| Implements Coupled Ailerons and Rudder (also known as “CAR”) | |
| Operation | |
| ◥ | Sets mode: |
| +/- | Cycles between OFF/ON and changeover switches S1 – S5, SI and L/S |
| R | Toggles between OFF and ON |
| ◢ +/- | Movement of slave relative to master 0 – 200% |
| R | Toggles master/slave mode AILERON→RUDDER and RUDDER→AILERON |
| Notes | |
| Cascades its output to other mixers, i.e. you set CAR on top of other mixing such as VTAIL or VTAIL+. | |

| Screen Name | Navigation |
|---|---|
| **Identify Widgets** | **Control→Test** |
| Description | |
| Reveals the codes for all the widgets (except dual-axis control sticks). Displacing a widget highlights the associated code. Use this screen to<br><br>Determine the codes of each widget.<br><br>Check switches and levers have been assembled the right way up. | |
| Operation | |
| ◣ | DigiAdjuster test. After pressing this key, “>” should be displayed if digi-adjuster is rotated clockwise, and “<” if anticlockwise. |
| Notes | |
| The display shows secondary switches (S1 – S5 and L/S) in lines 1 and 2, and stick/switches E – I in lines 3 and 4. Also shown are the Memory switch and Digi Adjuster. | |

Program Screens (cont)

| Screen Name | Navigation |
|---|---|
| **File Copy** | **Files→Copy** |
| Description | |
| Copy or erase a model memory | |
| Notes | |
| See *mc3030 User Manual (p. 47)*. | |

| Screen Name | Navigation |
|---|---|
| **File Name** | **Files→Name** |
| Description | |
| Change the name of a model memory | |
| Notes | |
| See *mc3030 User Manual (p. 47)* | |

| Screen Name | Navigation |
|---|---|
| **File Shift** | **Files→Shift** |
| Description | |
| Change the active model memory | |
| Notes | |
| See mc3030 Users Manual (page 50) | |

| Screen Name | Navigation |
|---|---|
| **Check Trims** | **Files→Chktrim** |
| Description | |
| Check the position of the slider trims | |
| Notes | |
| See *mc3030 User Manual (p. 51)*. | |

Program Screens (cont)

| Screen Name | Navigation |
|---|---|
| **Timer** | **Menu2→Timer** |
| Description | |
| Set up the stopwatch | |
| Notes | |
| See *mc3030 User Manual (pp. 16-18)*. | |

| Screen Name | Navigation |
|---|---|
| **Op Time** | **Menu2→OP.time** |
| Description | |
| Reset the operating time | |
| Notes | |
| See *mc3030 User Manual (p.16)*. | |

| Screen Name | Navigation |
|---|---|
| **Assign Control** | **Menu2→Assign→Control** |
| Description | |
| Assign a control to a stick or switch | |
| Operation | |
| +/- | Cycle over widgets A-I |
| +/- | Cycle over controls |
| Notes | |

Unused controls should be explicitly assigned to “UNUSED”. Otherwise they may be providing unwanted inputs to mixers.

If you assign more different controls to the same stick/knob only one assignment will actually work. Best avoided!

Rules for Assigning Controls

- *Rule One: You can’t assign the same control to two different widgets A-I.*
- *Rule Two: Do not assign different controls to the same stick/knob* (you are physically able to do this but the results are unpredictable).

Program Screens (cont)

| Screen Name | Navigation |
|---|---|
| **UserMix** | **Menu2→Assign→UserMix** |
| Description | |
| Define a user mixer. | |
| Operation | |
| +/- | Cycle between four mixers USR-MIX1 to USR-MIX3 |
| +/- | Cycle between four inputs for the selected mixer |
| +/- | Select the control to use as input. |
| R | Toggle between Normal and With Trim (identified by +T). |
| Notes | |
| If With Trim is selected the trim lever contributes to the mixer input signal (and hence all the outputs).<br><br>After you press M to finalise the mixer definition, you must assign (or re-assign) your servos to the mixer. When you assign a servo, the system alters the servo definition so that the correct inputs appear in the Servo Travel pages. Failure to assign the servo to the mixer means that any changes will *not* be reflected in the servo travel pages. In order to remind you of this, the system takes you directly to the Assign Servos screen when you press M to exist the UserMix screen. If you don't reassign your servos, the old (unedited) mixer definitions will continue to be used. | |

| Screen Name | Navigation |
|---|---|
| **Soft Switch** | **Menu2→Assign→Softsw.** |
| Description | |
| Define a virtual secondary switch | |
| Operation | |
| +/- | Choose master widget A-H. The corresponding control is displayed. |
| +/- | Choose the switching threshold |
| Notes | |
| Gx appear in the secondary switch lists as SA, SB, SC, SD, SE, SF or SG depending on the master stick/switch chosen. | |

Program Screens (cont)

| Screen Name | Navigation |
|---|---|
| **Pupil** | **Menu2→Menu3→Pupil** |
| Description | |
| Master-Pupil connection (a.k.a. “buddy box”). | |
| Notes | |
| See *mc3030 User Manual (pp.74)*. | |

| Screen Name | Navigation |
|---|---|
| **Teacher** | **Menu2→Menu3→Teach** |
| Description | |
| Master-Pupil connection (a.k.a. “buddy box”). | |
| Notes | |
| See *mc3030 User Manual (pp.75)*. | |

| Screen Name | Navigation |
|---|---|
| **RPM** | **Menu2→Menu3→RPM** |
| Description | |
| Rev counter readout. Requires MPX rev counter sensor. | |
| Notes | |
| See *mc3030 User Manual (pp.18)* | |

| Screen Name | Navigation |
|---|---|
| **Transmission Mode** | **Menu2→Menu3→PCM/PPM** |
| Description | |
| Select transmission mode. | |
| Notes | |
| If using a Futaba or other third party PPM receiver, select PPM9. See *mc3030 User Manual (pp.18)* for more info. | |