

LADDERING RESISTORS and The AUTOBRAKE

ian@737ng.co.uk



ian737ng



Right, I was looking for a solution for the Autobrake because the PMDG does not respond to normal FS Autobrake commands and eventually I came across two can you believe.

The most simple solution is via FSUIPC v3.7 onwards. In this version, Pete Dowson allows you to assign two functions to each joystick card axis. So if you have a TQ that gives you Reverse Thrust capability, then you can assign Reverser 1 and Left Brake to that lever axis and Reverser 2 and Right Brake to the other. Simple really because Reverse Thrust and Autobraking occur both at the same time in the landing roll and are both turned off at about the same time. And the second solution (which I have found to be more realistic because you have to set the Autobrake control) is to 'ladder' resistors across a rotary switch to govern the amount of braking pressure applied at any given position of the selector switch.

CONDITION

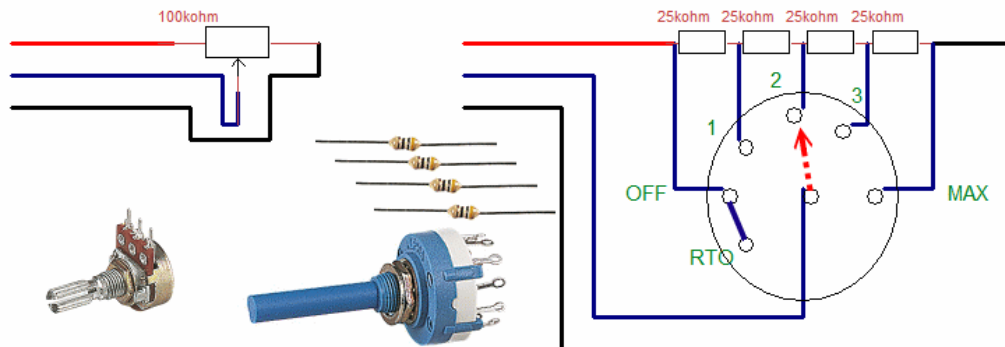
In the landing roll, automatic brake pressure is applied at a predetermined level dependant on certain factors like runway length, landing weight, runway surface condition etc. As the speed decreases to about 60kts, reverse thrust is returned to idle, Autobraking is switched off and manual braking is then introduced to take the aircraft down to taxi speed.

SOLUTION

A Rotary Switch was assembled with 25Kohm resistors between each switch contact. This allowed me to have an exact level of control over the amount of brake applied. The term used to describe this approach is 'Laddering'. I am forever grateful to Mark DePoneo for teaching me this.

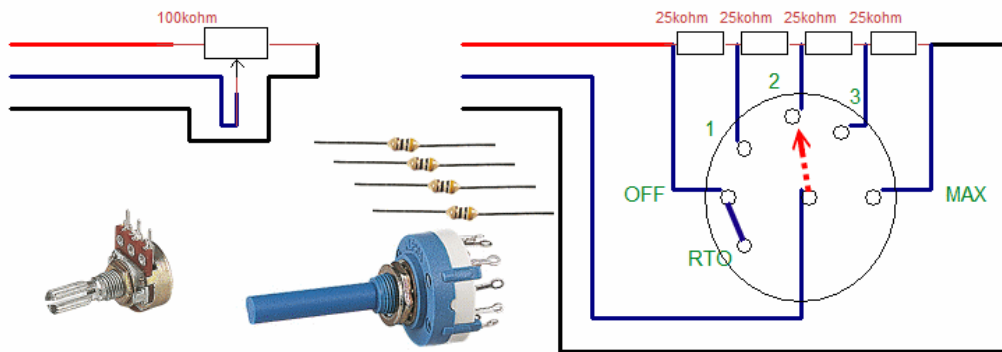
You will require a registered version of FSUIPC for this because once we have the switch wired to our joystick card or emulator, we will assigning the axis to brakes in FSUIPC. It's much more accurate than FS itself.

Ok, lets quickly talk about joystick inputs either thru a proprietry card or thru one



of the many emulators available like Opencockpits or Leo Bodnar. Any joystick axis is controlled by a resistance which changes. Most axes are controlled by variable potentiometers like the one on the left. This potentiometer will have a built in resistance. Usually they are 100Kohm (although it's not set in stone). You can use any resistance, but I use either 10Kohm or 100Kohm. Smaller ratings use more current and higher ratings make more noise or spikes. Your choice. What happens is that 5vdc is applied to one outside pin and Ground to the other outside pin. These pins are connected by (usually) a Carbon Track on which the wiper moves along. The centre pin referred to as the wiper moves along the carbon track and as it does so, the resistance between the two extremes of the track changes. It's a sliding scale, as the resistance changes, so does the voltage applied (ohm's law) . It is this change in voltage that the wiper returns to the joystick card as to it's position on the track. So for example, a 100Kohm variable resistor will resist from virtually nothing to 100Kohm as the wiper moves along the track - get it :o)

OK, instead of a carbon track to change the resistance between the 5vdc and Ground, with laddering we introduce known value resistors.



In the diagram above, these two controls are doing exactly the same job. They both have a total resistance of 100Kohm which is variable. But the control on the right gives 5 distinct resistance reports, thus 5 distinct settings on the range of action which is what I'm looking for.

Assign your axis in FSUIPC to BRAKES (there's a reason for this, I'll explain later). Look at the above diagram. If the red wiper is either connected to OFF or RT0, no braking pressure is applied except from the brake pedals on your CH (we've all got them haven't we).

Now as you rotate the switch clockwise, at pin 1, 25Kohm resistance has been introduced, thus your axis will invoke 25% of braking power. Go to pin 2 and with 50Kohm introduced, thus 50% of braking power will be introduced. At pin 3 75Kohm and 75% braking and at MAX 100Kohm and 100% braking power.

TIPS

If your switch works the wrong way round, simply swap the 5vdc and the ground and operation of the switch will be reversed.

Now the above example is 'linear'. That is to say at each increment of the switch, an exact addition or subtraction of the resistance is made. By changing the value of the resistors in the ladder, you can also alter the amount of braking applied. For example, you may want to put a 50Kohm between OFF and 1, a 33Kohm between pin 1 & 2 and 2 10Kohm resistors between 2 & 3 and 3 & 4. This way you will get a higher braking action at the bottom end of the range or swap them round for lesser action at the lower position numbers. Refer to Boeing documentation for the amount of brake pressure applied at any given selected position and have a play :o)

WHY I USE FSUIPC

I love this utility. It is far more accurate, reliable and stable than FS can ever dream of being.

Both my toe brakes have been assigned thru FSUIPC. Now I am introducing a third source of applying brakes to the aircraft. Having connected my Autobrake thru Leo Bodnars excellent BU0836 joystick emulator card, I have assigned the axis in FSUIPC to Brakes. Now going thru the landing checks, the Autobrake is selected at a given position (usually 2 or 3). As we land braking pressure is applied and reverse thrust is applied.. Braking pressure is still applied by the Autobrake until I touch the toe brakes, where the most recent input (i.e.from the toe brakes) becomes the governing axis of how much brake pressure is applied. Basically the toe brake application over-rides the Autobrake input, in effect disengaging it. I have put this thru a couple of days of testing (when the bride says it's ok) and it works perfectly everytime. It's a very realistic solution and I'm going to stick with it.

As always, if there is a better way to do it, tell me, I'm dying to know. And as always thank you for your interest and feedback – vitally important.

Regards ... Ian

Oswestry, England 29/11/06

