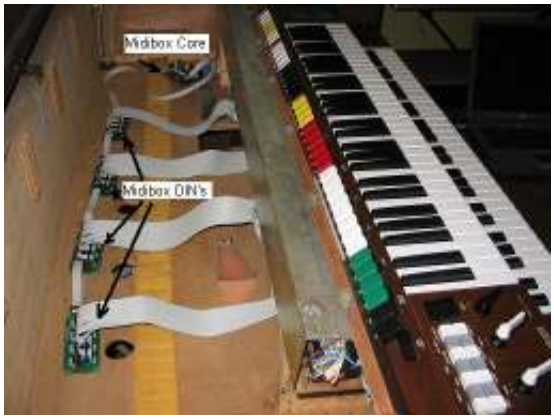
Midification of an Organ

MIDIfication is a new word that has been coined to describe the process of adding the MIDI communication function (Musical Instrument Digital Interface) to an older electronic organ. http://en.wikipedia.org/wiki/Musical_Instrument_Digital_Interface . It is an interesting adventure that can vary in cost from a couple of hundred dollars (plus many hours of time) to many thousands of dollars (and not as much, but still a lot of time). [What else would you rather be doing?]

To commence the MIDIfication of an organ you must be prepared to use a soldering iron. If you have never attempted this type of thing before, it would be best to organise some help from a friend, an organ society member with some experience, or engage the services of an organ repair technician. Help in the early stages will be very useful, and you may gain confidence to proceed on your own once you have mastered the basic skills.

You can chose to purchase ready made modules to connect to your organ (eg. Classic Organ Works http://www.organworks.com/Web/about/index.asp) or decide to build your own in kit form (uCaps Midibox) http://www.ucapps.de/ . There are of course other options you could research yourself. In evaluating other options, check the ability to program extra MIDI functions (other than just Note ON/OFF) and the ability to add multiple systems for expansion of extra keyboards, piston and/or stop tab functions. Either way, every keyboard note, pedal, piston (and even stop tabs if you want) will need to have wires connected from them to the MIDI modules. (This could be in the vicinity of 250x wires and connections.) This article will give a detailed overview of building and installing the uCaps MidiBox system.

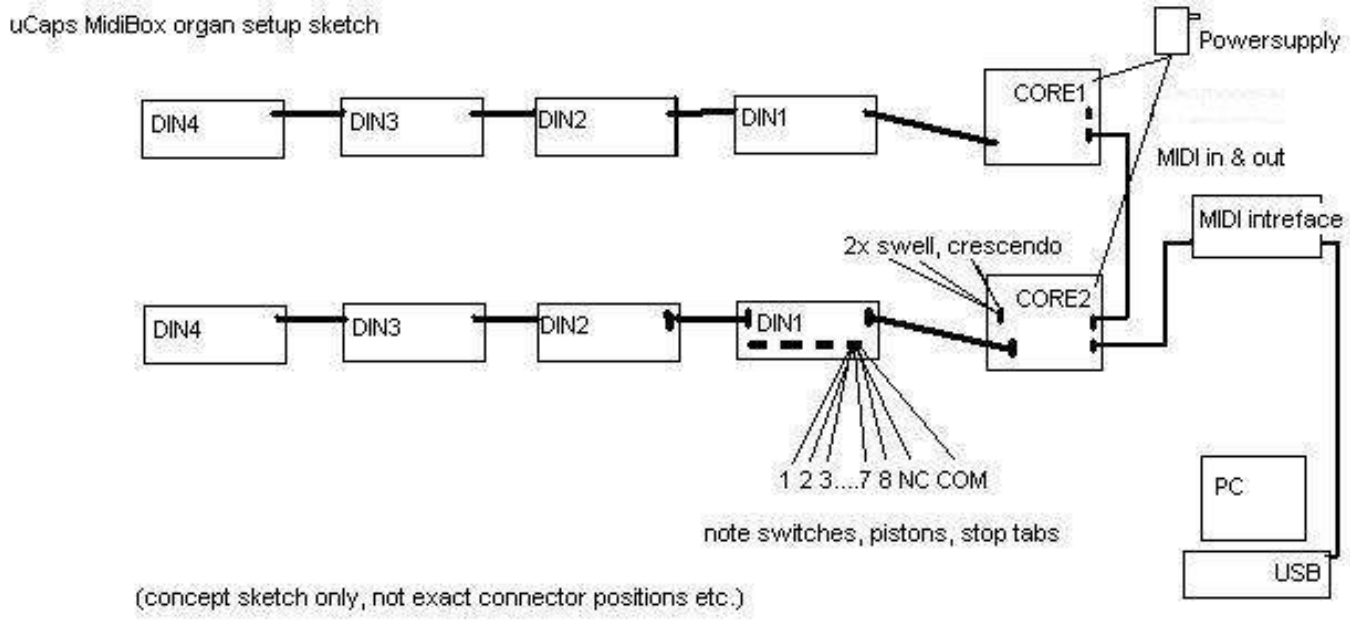
uCaps MidiBox is an open source concept using a PIC chip Microprocessor to generate and process MIDI signals from events such as keyboard notes and pedal movement. The software code and the hardware design is freely available to the "user-community" to use and modify as each application requires. Modification and improvement of the system is given back to the "user-community" for all to share. No one is permitted to commercially profit from the designs. The community has approved a couple of individuals to make available printed circuit boards and component kits to help hobbyists get started on their projects. Smash TV http://www.avishowtech.com/mbhp/buy.html is one of these suppliers.



First you need to decide if you want to keep your old organs original sound function or just use it as the physical console for your VTPO. This will depend on how much money or emotion you have invested in your organ. With later model Conn organs there are spare switch contacts on every note so adding MIDI without affecting the original electronics is easy. With other organs you may need to add switches or magnetic reed switches to every note if you are determined to keep the original electronics. This will add cost to your project. (I picked up a Yamaha E5 series organ as a "give-away". It had many compounding electronic faults so I decided to strip out the entire electronics).

The next decision is how many keyboard notes, second touch, pedals, pistons, stop tabs, swell & crescendo you want to link to MIDI. (Keep in mind at this stage you can use inexpensive Casio or Yamaha type keyboards with Midi already operating to reduce the scope of your project, but then you need to build a stand arrangement to position them stably) you may then only need to "MIDIfy" pedals and/or pistons. OK, so as an example, you decided on 3 x keyboards (3 x 61), 1 x 32 note pedal board, 38 x pistons, 6 x toy counter, 2 x swell and 1 x crescendo. (61 + 61 + 61 + 32 + 38 + 6 = 259)

So what MidiBox hardware do we need? The MidiBox system uses 2 x types of hardware circuit cards, CORE and DIN. The CORE houses the PIC processor chip and is where the MIDI cable connects to your MIDI to USB interface. Each CORE takes up to 4x DIN cards and also accepts 3 x analogue inputs for 2 x swell and 1 x crescendo. Each DIN card takes 32 x switch connections (so each CORE takes up to 128 switch connections). With the example above we have 259 switch functions we want to connect. Divide 259 by 32 to find out how many DIN cards we will need (259 / 32 = 8.09…). Decision time! For this example we will need 9 x DIN cards (with 4 x DIN cards per CORE) we will need 3 x CORE cards. (If we reduce our project by 3 switch functions we can do it with 8 x DIN cards and 2 x CORE.) There is no limit to the number of CORE cards we can use, and every switch function if freely programmable to any of the 16 MIDI channels and any MIDI function, so there is no problem with duplication of event. (With the spare capacity on your third CORE you may decide to add stop tabs in the future, or you may decide at this stage to just use 2 x CORE and 8 x DIN (and not install 3 x of your pistons).



Now you can order your kits from Smash TV. Please note, when ordering CORE kits you should request that they have the bootstrap loader installed and that each have a different address number (ie. 0, 1, 2).
The bootstrap loader means you can load the software for your project through your MIDI to USB interface cable and so don't need a PIC chip programmer device. The different address numbers allow you to link more than 1 x CORE to 1 x Midi to USB interface, but more to this a little later (see MIDI MERGE in MIDIO128 section).

Other things you will need. You will need a power supply to power each of the CORE cards. This can be an inexpensive plug-pack power supply. (If you are not confident to select a power supply, ask your support person). You will need 10 way ribbon cable and 10 way header plugs to connect the DIN cards to the CORE and the keyboard switches to the DIN cards. The length of cable you will need will depend on the size and layout of your organ. Often it is difficult to decide how you will rout the wiring till you make a start on it. (For the example above you will need 48 x 10 way header connectors, [I suggest you get a handful of spares] and I would suggest about 20m of 10 way ribbon cable) I actually used 50 way ribbon cable and was able to group 4x 10way strips per DIN card on the keyboards to keep it neater and use the extra 10 way left over to join the DIN cards. It is best to layout the order of your wires in the neatest logical physical layout. Correction of the order electronically is all done in your custom software configuration (explained below).

You can choose to add an LCD display. Without it there is nothing to show you if anything is happening other than seeing if anything is happening on your PC program. (See MidiOX below in the software section)
The PCB component layouts and assembly guides are comprehensive on the uCaps and Smash TV web sites, so I won't cover that here.

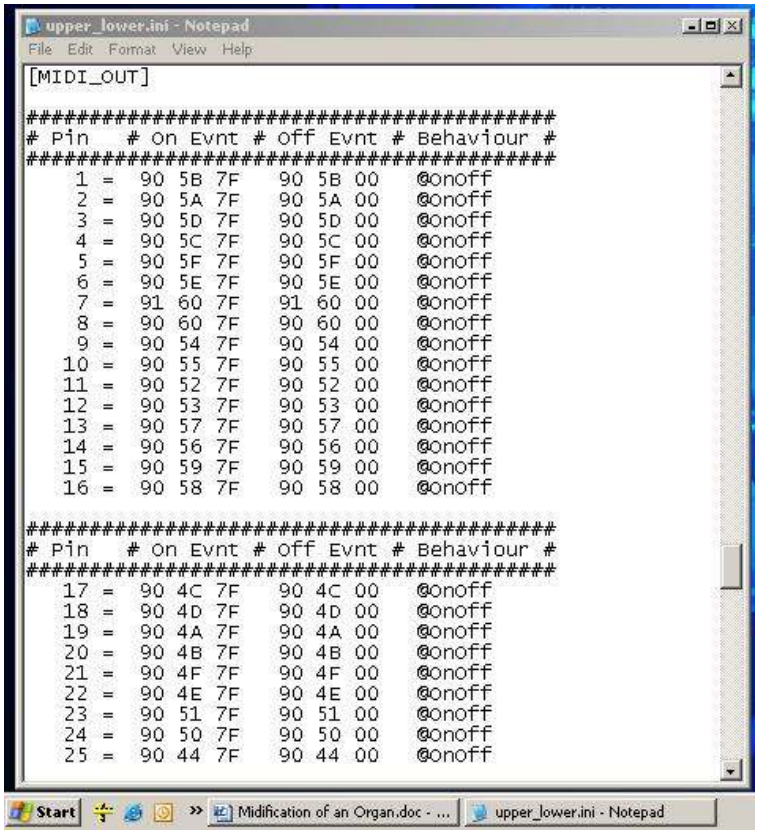
Once you have ordered your hardware you will want to think about the software aspect of the project. Again the details are on the uCaps web site, however in the interests of giving you a clear picture of what is involved, I will give an overview of the steps involved.

All of the software involved is free and downloaded from the uCaps web site or from links given on the site. I mentioned earlier that you order the PIC chips with the "Bootstrap Loader" installed on the chip. (Smash TV supplies them this way.) This makes it possible for you to load all the relevant software through your MIDI Interface connection. (If you thought you could get the PIC chip cheaper through a different supplier, you would need a PIC programmer interface and then need to load the "Bootstrap Loader" yourself. This adds cost and complication you don't need to add to this project.)

First you need to load MIOS into each of your CORE modules. To do this you use a program called MIOS Studio. (This program is written in Java so you need to load the latest version of Java into your PC for it to run.) (MIOS is also now loaded standard by SmashTV)

Next you need to load MIDIO128 into each of your CORE modules. To do this you use a program called SYSEXBOX (Can now be loaded using the latest MIOS Studio). (This gets you to the point where your DIN cards will work if you connect a common to one of the input pins. The MIDI codes sent out at this stage are default codes so won't give the correct notes out but at least you will see some activity.) To link multiple CORE modules to 1 x MIDI interface you need to activate MIDI MERGE by modifying the MIDIO128 file. There are special versions of this file to activate the swell and crescendo functions available from contacts on the Miditzer Forum.

The final step is to load MK_MIDIO128_SYX. Again this is loaded using SYSEXBOX. This is the file you modify to get the Midi code you need from each of your switch contacts. You make the changes simply by editing the preconfigured file using "Notepad" (the simple text editing program on all computers). Once you have made your changes, your file needs to be fed through a program called "Active Perl". This is a program you load but never need to run yourself. There is a pre-configured batch file in the MK-MIDIO128_SYX folder that does all the work. At this stage of your task, it is useful to see on your computer what MIDI signals are coming out. The MIOS Studio program mentioned above will show you the MIDI codes as they come from your hardware.



Another program called MidiOX http://www.midiox.com/index.htm is another free program that will show you the MIDI signals. (It can be useful to connect a factory MIDI device like a Casio or Yamaha keyboard to see what note numbers should be. It is also useful to learn some basic MIDI codes and have a DECimal to HEXadecimal conversion chart)

Now you have finished! You may never need to touch the system again, however if you ever make hardware changes or additions you can reprogram any parts of it or all of it, all over again as many times as you like.

Written by: Rick Whatson – TOSA Queensland Division – Updated July 2009 greenfox4075@gmail.com