Expanding the expressive potential of MIDI keyboard instruments

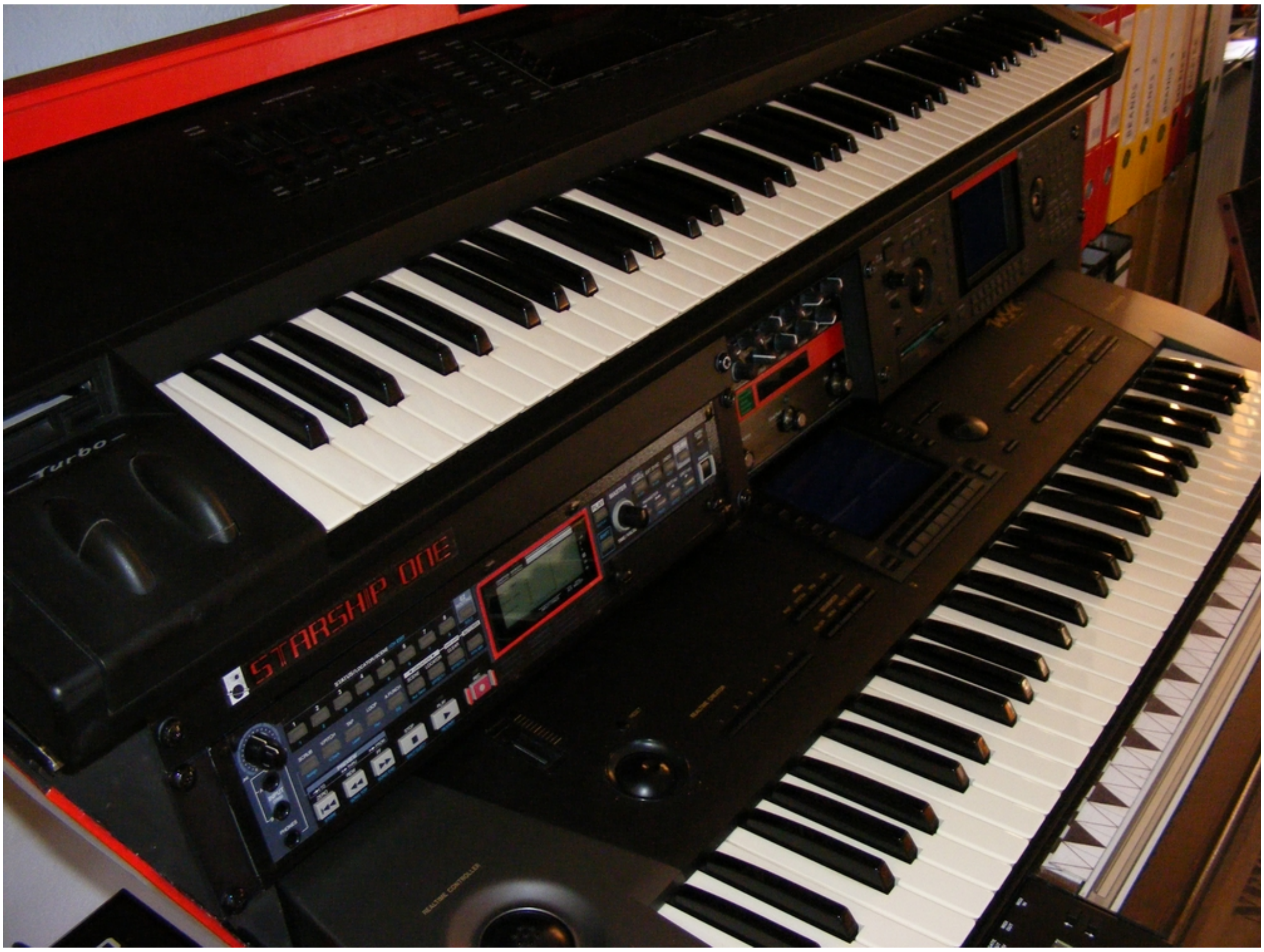

Fig. 1: Oh no, NOT that Starship One contraption again! Sorry guys, It has by now become the main focus of a whole set of articles.

by Marc Brassé

1 Introduction

One of the most revered Synthesizers of all time, the CS80, is famous for its sound but also for it´s velocity and poly aftertouch sensitive keyboard and it´s ribbon controller. It offers a depth of expression that few other instruments, including ¨real¨ acoustic ones, can offer.

More instruments have provided such facilities in the past. The Sequential Circuits Prophet T8, many Ensoniq synthesizers and the GEM S-series workstations provided poly aftertouch and a reasonable number of stand alone MIDI keyboard controllers offering it have also been produced (Roland A-series, Elka MK 88).

Kurzweil has acted as a typical promotor of integrated long travel ribbon controllers but stand alone ribbon controllers have even been available for a much longer period (Moog, Doepfer, etc.)

No instruments have however combined both features so effectively in one package as the CS80 did.

Such facilities actually never really caught on. It is one thing to hammer out a decent melody on a piano keyboard but producing real expressive results is yet another thing. First of all one has to be able to imagine what more could be possible and secondly one has to build up the necessary skills to translate these idea´s into music. In that respect developers of expressive controllers have always had a hard time. How to offer the ultimate in control and still be accessible to the novice is a spread no system has yet been able to make.

A new renaissance seems to be dawning though. Technologically speaking the present situation surrounding musical controllers is actually better then it ever was. One can buy multitouch sensitive controllers such as the Hakan Continuum, the Roli Seaboard and the Linnstrument. On these tools one surface actually provides multidimensional control of several parameters such as velocity, pressure and finger movement per note!

So the sky has become the limit at last?

Well, not quite yet. If one is a fan of fully programmable hardware solutions one still tends to be left out in the cold. Often a computer is needed to interpret all the generated control data, even when the sounds are actually produced by hardware.

Sounds like double trouble, eh?

Furthermore it is not actually likely that the dominance of the conventional keyboard will be broken on the short term. To many vested interests will work against it. Didn´t you yourself for instance not already go through enough trouble to learn to manage it yourself?

To get to the core of the matter in hand: This article describes a method to get the best expressive potential out of existing multitimbral MIDI keyboard instruments.

The examples below will be focused on the use of the Doepfer R2M ribbon controller and the MIDIMIXFIX Midi data management device driving any multitimbral MIDI hardware or software synthesizer, in this case represented by the Technics WSA1 synthesizer.

The general approach will however also be of interest to users of other MIDI devices, since the content also hits on the general complications one might stumble over when trying to make different MIDI components communicate optimally. So even if this text does not provide the easy way out to address problems associated with a specific tool, instruments or setup (there are SO many possible permutations out there!) it might still point towards a fitting solution for any other specific MIDI related problem.

2 Ingredient A: The necessary mindset

First things first though.
One has to be quite aware that the trigger oriented culture in modern electronic music has a clear reason. Most people that make electronic music are not skilled musicians. In itself there is nothing wrong with that. The more creative people the merrier. Talent is not only expressed by playing difficult riffs and chord progressions. The move away from training however also poses a certain danger. Nowadays the machines often dictate the music that comes out of theme more then their users do.
As already mentioned in the introduction one however has to think like a skilled musician if one wants to get the best expression out of electronic instruments. In other words: One has to learn to play the device in question, however it might be configured.
Important: I mean the word instrument as wide as possible here. Any specific setup of music production tools including its input devices will already be such an ¨instrument¨. That does not necessarily mean that one needs to receive a whole classical musical education. It does however imply that one has to understand his ¨instrument¨ to his instument well enough to circumvent its limitations since every instrument has such limitations.
In short: Even the best controller will do very little for you if you do not practice using it.
Let me give an example: To get the best out of the Doepfer R2M ribbon controller described below we actually have to avoid using most of the automated tricks built into it´s software to make the average user´s life easier. In other words: Ones mindset has to be that of a true performer and not that of a master of ceremonies or typical button pushing, mouse wielding DJ.

Fig.2 : The GEM S3. A much underrated instrument. Within the context of this article it represents the rare species of the poly aftertouch capable keyboard synthesizer / controller.

3 Ingredient B: The Controllers

Controller-wise we will concentrate on 2 devices here:

1. The GEM S3, representing all poly aftertouch capable keybaord controllers out there.
   As such we can be rather short about it. It generates poly aftertouch and is darn good at it. Of course we can moan about the (marked) difference between the resolution of its internal processing of control data to the crude way MIDI transmits such information but that is not the fault of the S3. The MIDI standard simply does not offer the ultimate in controller resolution. For now we will however just have to live with it. Full stop!

2. The Doepfer R2M, representing all stand alone available ribbon controllers and multidimensional controller surfaces
   The R2M ribbon controller has the same response problem but, boy, is it a multifaceted thing. It can generate a multitude of MIDI commands with ust one single touch. It even has a nice integrated programmable arpeggiator. In other words: It´s software is very well suited to the present generation of trigger addicted musicians.

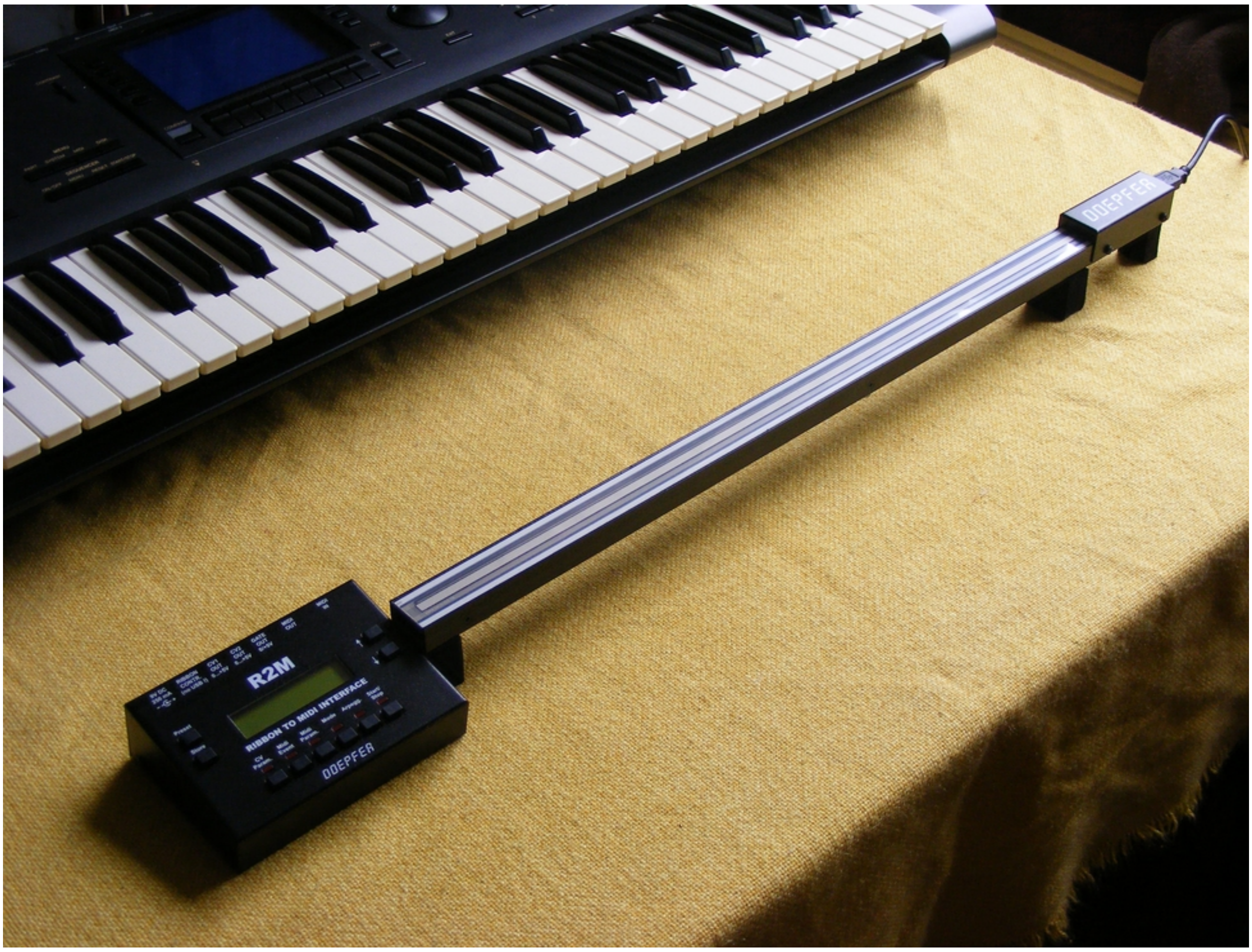

Fig. 3: The Doepfer R2M ribbon controller, exisiting of the actual ribbon controller (at the right) and a separate programmer box with MIDI and CV outputs (at the left)

But wait a minute. Didn't the R2M actually come about as part of a project to build a more modern electronic version of the Trautonium?

The what?

A little bit of history might be in order. In spite of most people thinking that Bob Moog invented the synthesizer the truth is actually much more complex and interesting. In the past 150 years or so a lot of extremely expressive instruments where actually developed. We'll not go into the whole story here but only mention that in the German Professor called Ludwig Trautheim constructed an instrument on which pitch could be continuously controlled through a long ribbon, thus giving the instrument the expressiveness of a bowed string instrument. This tool was later developed further by it's most avid user, Oskar Sala. He is most known for his contribution to the Soundtrack of Alfred Hitchcock's The Birds but he was actually a very much broader musician and composer.

To cut a very long story short.

German modular synthesizer builder Doepfer later worked on a project to build a more up to date interpretation of the original instrument. One of it´s offspins was that the R2M ribbon controller. It can provide the same functionality that original Trautonium ribbon manuals offered by generating MIDI and CV information from a 50 cm long ribbon which also reacts to pressure, the most obvious choices for both being pitch and filter control respectively.

4 MIDI and pitch / pitch and MIDI

OK, whatever. I'll buy an R2M, get it out of its box, fire it up, put it in it´s Trautonium mode and Bob's your uncle.

But why does the synthesizer one attaches to it then still react so stubborn? Why does everything feel so unnatural and counter-intuitive. That´s not what we bargained for!

Well, for optimal results you first have to teach both machines to properly communicate with each other.

First you have to match the pitchbend settings at both ends. Please understand that almost all MIDI values are relative. MIDI only provides a certain range of steps (normally 127 at best). How those values are matched is however left to the units sending or receiving that info. That might seem strange at first but it actually makes a lot of sense. It means that you can dramatically improve the specific way the sending and receiving units react be carefully configuring the way they communicate.

Concerning the R2M's behavior when creating MIDI pitch information: Initially only a note number is produced and the pitchbend info is then added when one modulates the signal with further movements of the finger. For instance; assuming you have pitch routed to the ribbon length (so sid eot side, left to right) you will first get a pure note and then a pitch bend number will be added after first repositioning your finger. In other words: The initial touch only generates a note number and and added left to right movements then lead to pitchbend info.

In other words again: You have to be aware that MIDI can only create pitches between the normal 12 pitches in our western musical system if extra bend information is added to the note number. So any such pitches are always a product of 2 values.

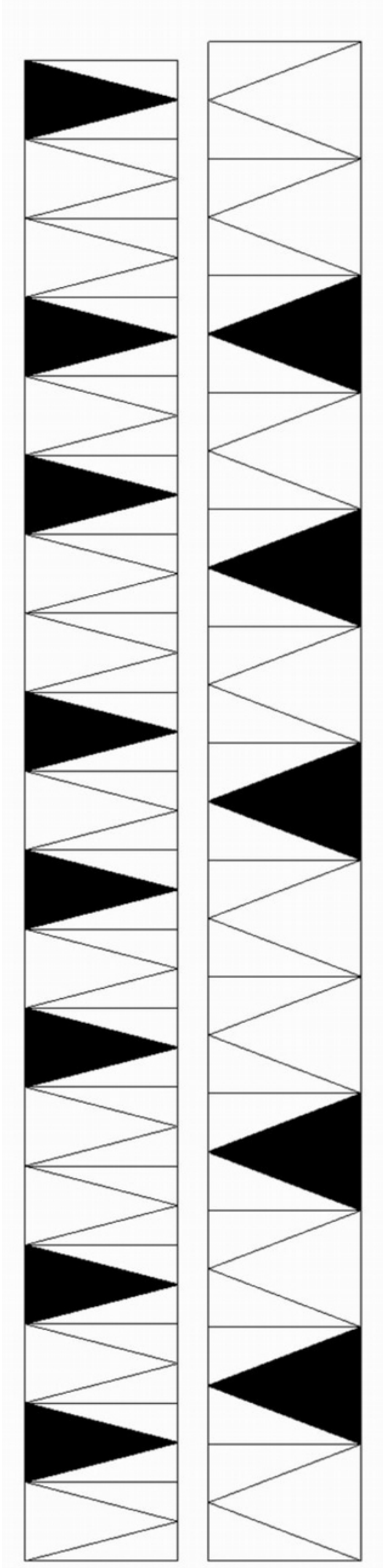

Fig. 4: Wait a minute! What piece of Neo-Constructivist s(beep) is this?

5 Ingredient C: Scaling it

If one tries to play the R2M in Trautonium mode one becomes aware of a rather frustrating problem. If one repeatedly triggers a note at almost the same spot the R2M sometimes behaves a bit erratic, as if it cannot decide what note to trigger. I have however found that the nearer to the pure pitch a note is triggered the less severe the effect is. So it is still important to put your finger exactly on the right spot. For this reason I have designed a scale with arrowed “keys¨ that enables me to see where I am.
I have created a special JPEG containing the keyboard designs for both the 2 and 3 octave layouts. If you make sure these are printed at the true scale contained within the file everything should more or less fit the R2M.

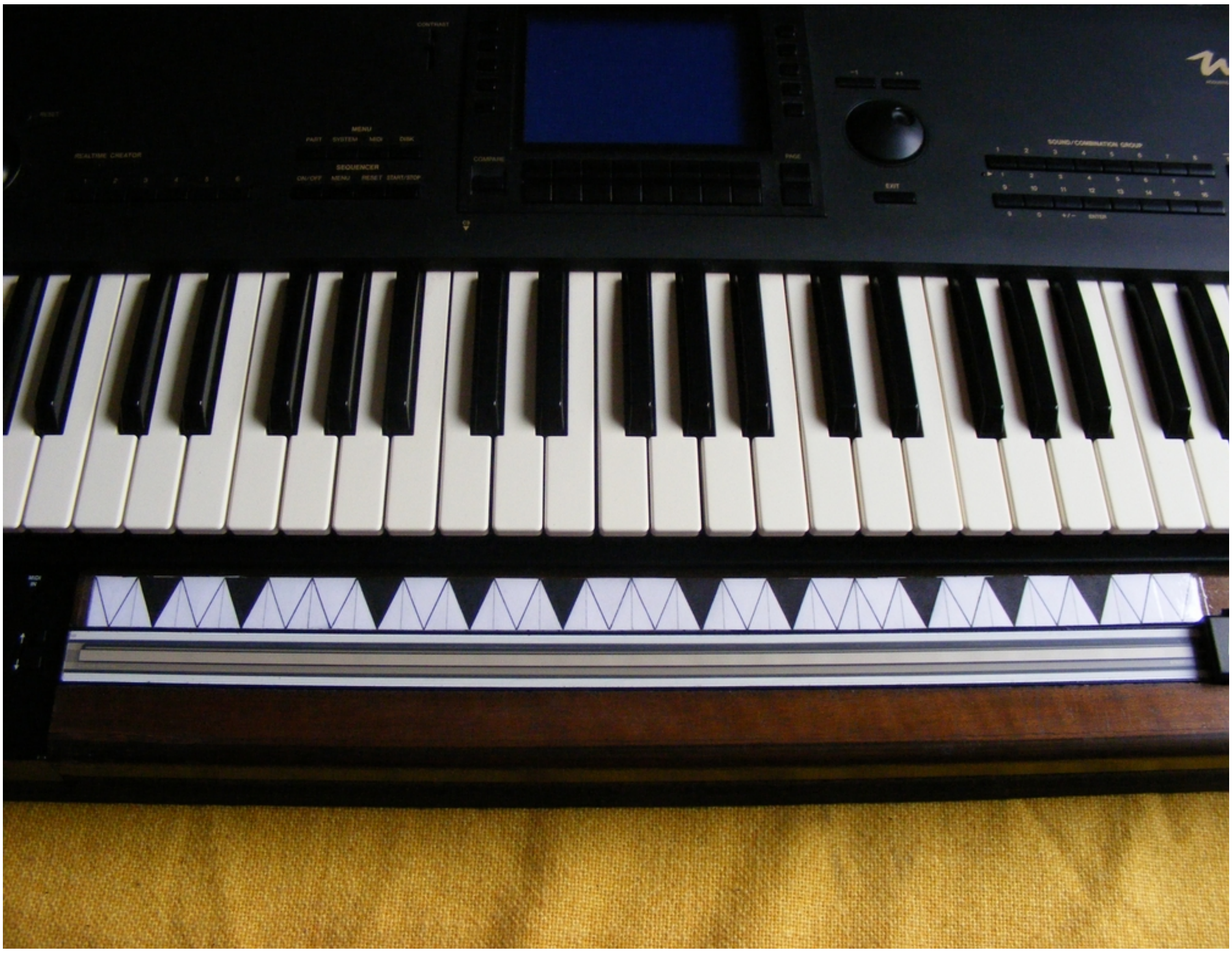

Fig. 5: So you want to know where to put your fingers, eh?

But where to leave that darn scale? There is no room to place it on the ribbon controller itself!
Some of the design decisions I described in my earlier ¨Pimp Your Ribbon Controller¨ article (see the reading list in paragraph 13) now also move into focus. Together with integrating the controller box into one unit, the extra surface above the actual ribbon controller now becomes more then a mere convenience. Or can you come up with a better place to put that keyboard scale?

6 Setting up

- First the pitch bend range on the receiving synthesizer has to be set. I typically put this on 12 semitones. That way one can bend a whole octave up or down from the triggered note number. On a pitchbend wheel this is of course a rather extreme setting but since we are using a long ribbon controller such a range is quite practical. One will then be able play complete sliding melodies as long as one stays within a +1 and -1 octave range from the pitch of the triggered note if one wants to avoid having to trigger a next note (Instant Theremin stuff!). Most instruments are actually not able to react to pitch info over a longer range. OK, we are still falling short of totally recreating the Yamaha CS80´s implementation (also 1 octave up but much more when pitching down) but at least we´ll get rather close.
- When setting up the R2M as a pitch controller you first have to decide how many octaves you wan to play. I normally use 2 octaves because that seems to be a good compromise between side to side movement resolution and total range. Setting up 3 octaves gives you a bigger range but it also increases the sensitivity of the system to tiny movements, thus increasing the chance of hitting the wrong notes numbers or unintentionally adding pitch info, so hat option is probably better used by real violin players.
- I'll not go into all the details of setting up the R2M here. Please read the factory user guide for all it´s typical setup info and terminology.
- These are however the pitch translation settings I normally use on my R2M / Technics WSA 1 combination:

  2 octaves:

R2M pitch scale setting	123
WSA bend depth setting	109

  3 octaves

R2M pitch scale setting	121
WSA bend depth setting	115

  One would actually expect that these values could stay the same for both ranges but by finely adjusting them one gets the best results. When I now bend an octave, release my finger and then put it down again a note is generated that starts at the same pitch I just let go off. In other words: The pitch of the new note (which is generated without pitchbend info) is the same as the one I just generated by bending a note that actually started one octave higher or lower. In other words although they generate the same pitch both MIDI info ¨packages¨ are actually quite different.
  An example: playing a C1 and adding a pitch bend value of +64 by sliding up a whole octave should generate exactly the same pitch as directly playing a C2.

7 Fine-tuning on the fly

These are the main points you have to take into account when fine-tuning the setup:

- The pitch indicator scale is not permanently attached to the R2M. The ribbon is in itself an analog device and its output thus tends to drift a bit during use. One can however ¨fine-tune¨ the system by moving the drawing according to the pitch ribbons behavior. One could therefore call this simple solution an “integrated trimming feature”.

- For practical reasons I glued glued the printed scale on a thin metal ruler. This gives the scale enough weight to stay in place and ensures proper alignment over the whole length. Any wrinkling will after all influence the precession of the derived playing!

- Try to find the spots where erratic jumping after initial triggering does not occur and align the specific arrow to this position.

- This optimal positions should for all notes should be the same for all the arrows, from the lowest to the highest note on the scale, without any need to repositioning the scale in between. If not your calibration settings and the size at which you printed the scale do not match!

- Because of the analog nature of the sensor it might take some time before full stability is reached. In itself this is not strange. When you fire the R2M up the ribbon has a certain ambient temperature. During playing you however warm it by introducing the warmth in your fingers but also because you are constantly rubbing it. This should however sooner or later lead to an average temperature somewhere between the ambient and your body temperature. Luckily my experience is that the scaling of the trigger points on the ribbon controller tends to move as a whole, so realigning your printed scale should be enough for intermediate corrections.

This might all seem to be a bit of a hassle but you should compare it to handling a guitar or violin. The tuning of a such instruments is just as temperature sensitive as that of the R2M. It's the trouble one simply has to go through to be able to precisely generate pitch (info) oneself and thus part of the true performers mindset mentioned in paragraph 2. It all needs practice but the reward is a level of expression which no normal keyboard can provide.

8 The R2M's software limitations

Are we there yet? Does the R2M now at last behave like the perfect musical controller? Not quite. There are actually a few limitations within the R2M's software that proof that the concept has not yet been taken to it's ultimate implementation. I have contacted Doepfer about this but they are in the typical spread a small company often finds itself in. Of course things can be improved but what about selling the existing stock off first. And if most users are perfectly satisfied with all these trigger options why rock the boat for a few wannabee violinist?
So please read the following not as a dismissal of the R2M and / or Doepfer. Look at it all as a road to even more insight.

1. We have talked about setting up the pitch over the length of the ribbon and using the pressure sensitivity to generate other commands, such as aftertouch. But what about velocity?
   Sadly the R2M does not generate velocity. The best way to imitate velocity is to indeed use the ribbon to generate aftertouch to control filter cutoff or volume (or

even both). Then one can simply trigger the note and add all further information immediately after that.
There is a catch however. The developers have decided to alwys add a velocity value of 127 to every note-on command.
On first sight that seems sensible enough. A velocity value of 0 would immediately make the synthesizer fall (almost) silent. But there is a catch here. If I want to control note expression with pressure but the note is already triggered at it's maximum very little room for expression is left.
Furthermore all the sounds already in your synthesizer will normally be optimzed for used at lower average velocity levels. So implementing a lower velocity value would have been much more practical. I find that if normally play velocities somewhere between 70 and 100 so an average around 80 would have been a better choice.
Better still would be the implementation of an adjustable velocity value, in other words, if one could tell the R2M what velocity value to generate. A rather simple bit of reprogramming could do the trick and let's hope that Doepfer will sooner or later find the time to make this a standard feature.

2. If one tries to use both the R2M and a standard MIDI keyboard to control the synthesizer something else occurs. As long as one plays on the R2M only everything goes right most of the time but as soon as one switches between th keeyboard and the ribbon or vise versa there is a very big chance that one will find that the keyboard suddenly seems to be offset. In other words. It does no longer produce the pitch fitting the key one plays. There is a simple reason for that. The last note that was generated with the R2M ended on a certain pitchbend value. The synthesizer now thinks that value is still relevant and thus adds it to the note value you generate by pressing the next key(s).
The only way to get rid of the false pitchbend information is to quickly apply a bit of pitchbend via the keyboard's pitch wheel. The pitchbend value is then reset to 0. But that is a bit of a hassle, now is it? One wants to use both the ribbon and the keyboard because both have their own advantages. The ribbon controller might give perfect freedom in pitch but the keyboard can still add immediate perfect pitch triggering and polyphonic playing. In more practical terms: If you want to switch form a violin type controller to a chromatic one on the fly you should be able to that right away.
Have those Doepfer guys been totally daft then? No, not really. There is some sense in their thinking. What would happen if the R2M would generate it's own pitchbend reset message when your finger leaves the ribbon? Nice idea but every note with added release would then jump back to it's original pitch before it fully disappears.
What one would want is that a pitch bend to 0 message is generated every time one plays the next note and / or switches back to the regular keyboard.

3. And what about using the R2M as a dedicated pitch bend controller only, so in constant conjunction with a keyboard? Then one can really move into Vangelis style Yamaha CS80 pitchbend magic.
Again the velocity and pitch info generating functions of the R2M get in the way. The note number generation CANNOT be suppressed and we already spoke about the maximum velocity value. Again this could easily be fixed with a Doepfer software upgrade but new software cannot be loaded by the users so we will (again) have to wait for a factory upgrade.

9 Ingredient D: The Translator

Do we simply have to live with these limitations? Is there no alternative?
Well there is, although it means you will have to add a new element to the mix again, eg. buy an extra device, though not an expensive one.
The best way to fully open up the R2M's potential is to add something like the **MIDIMIXFIX**, developed by Danish software guru Lars Ole Pontoppidan.
The MIDIMIXFIX is a stand alone MIDI device that needs no laptop to program or control it. You just put it between your existing MIDI devices and it will filter out all your MIDI problems. That might not seem to be a big thing in an age where most musical needs are already handled by PCs, laptops and IPhones but some of us are still eager to actually make music instead of hassling around with instable platforms or consider NSA conspiracy theories.
Indeed the MIDIMIXFIX can solve all the problems mentioned above. But it can even do a lot more!

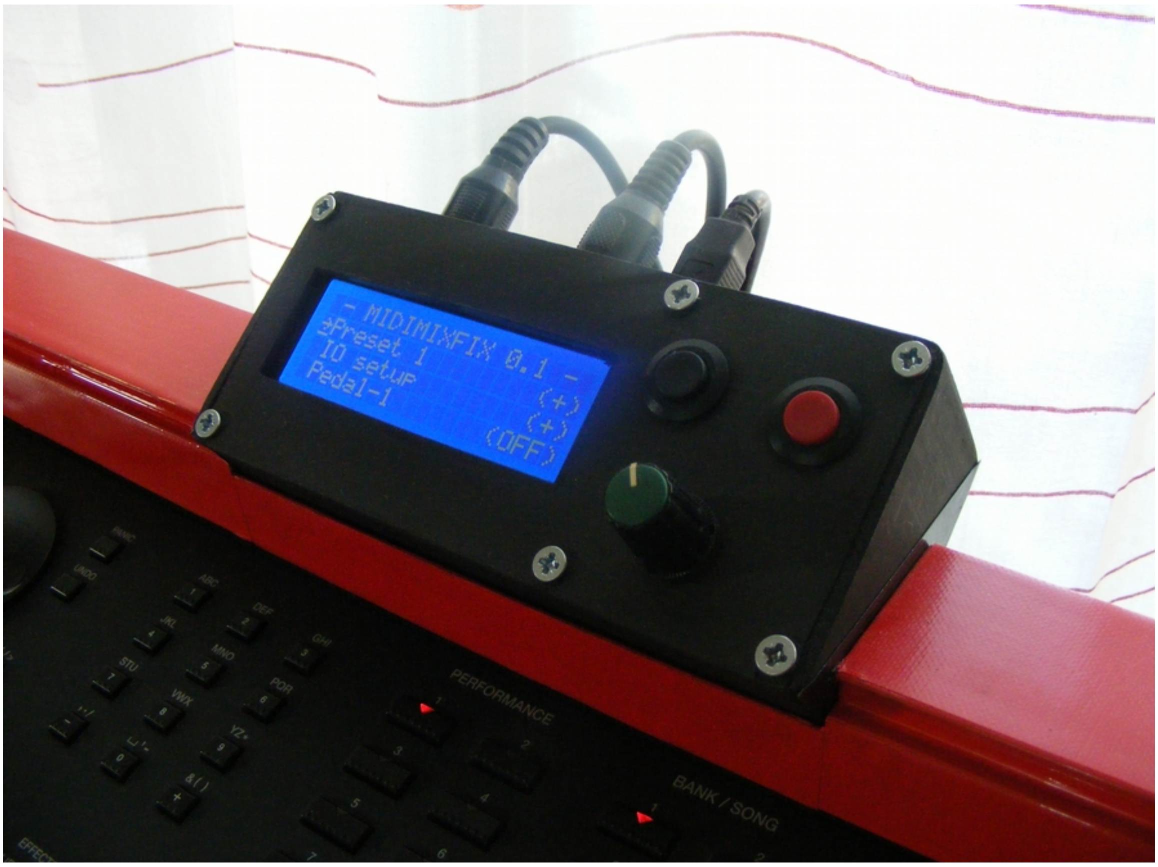

Fig.6: Call me a nifty little box or shoot me. It´s the MIDIMIXFIX!

10 Enter Poly Aftertouch

Why do I insist so much on poly aftertouch as being the ultimate controller?
Well, Because it is the one predetermined within the MIDI configuration to give every note you play on the same channel its own specific behavior. Most modern synthesizers are however not able to interpret such data. In other words: They listen to controller

information in a much cruder way. Any other controller info you send is interpreted as a channel specific command. So if you move your pitchbender everything you play (on the same channel) will be bent exactly the same way. Aftertouch then also becomes mono, which is nowadays therefore called exactly what it is: channel aftertouch

We are however not totally out of luck. Many an interesting synthesizer from the 1980ies onwards has a so called multitimbral mode, meaning that it can process MIDI data over multiple channels and attach a specific sound to each of these, enabling such machines to play complete arrangements.

What does however happen when we attach the same preset sound to all the individual channels? Then the synthesizer will play the same sound on whatever channel it receives the note data and thus (seemingly) mix eveything back into a single performance.

OK, at first sight that might seem to be a stupid thing to (why buy a multitimbral instrument at all then) do but if we can also get the controller keyboard to send control data per channel we suddenly end up with an instrument (again in the widest sense of the word) that can react to modification commands PER INDIVIDUAL NOTE. For conventional keyboards that is normally one step too far but if we can translate poly aftertouch into multichannel info our typical multitimbral synth can listen to that.

Logical enough isn´t it? So logical one would expect that many a keyboard controller / MIDI device will be able to offer exactly that.

The strange thing however is that hardly anybody in these preprogrammed trigger oriented times of ours seems to care much for it. Luckily for me / us MIDIMIXFIX developer Lars was prepared to teach it this very trick, a feat for which he has gained my ever lasting admiration.

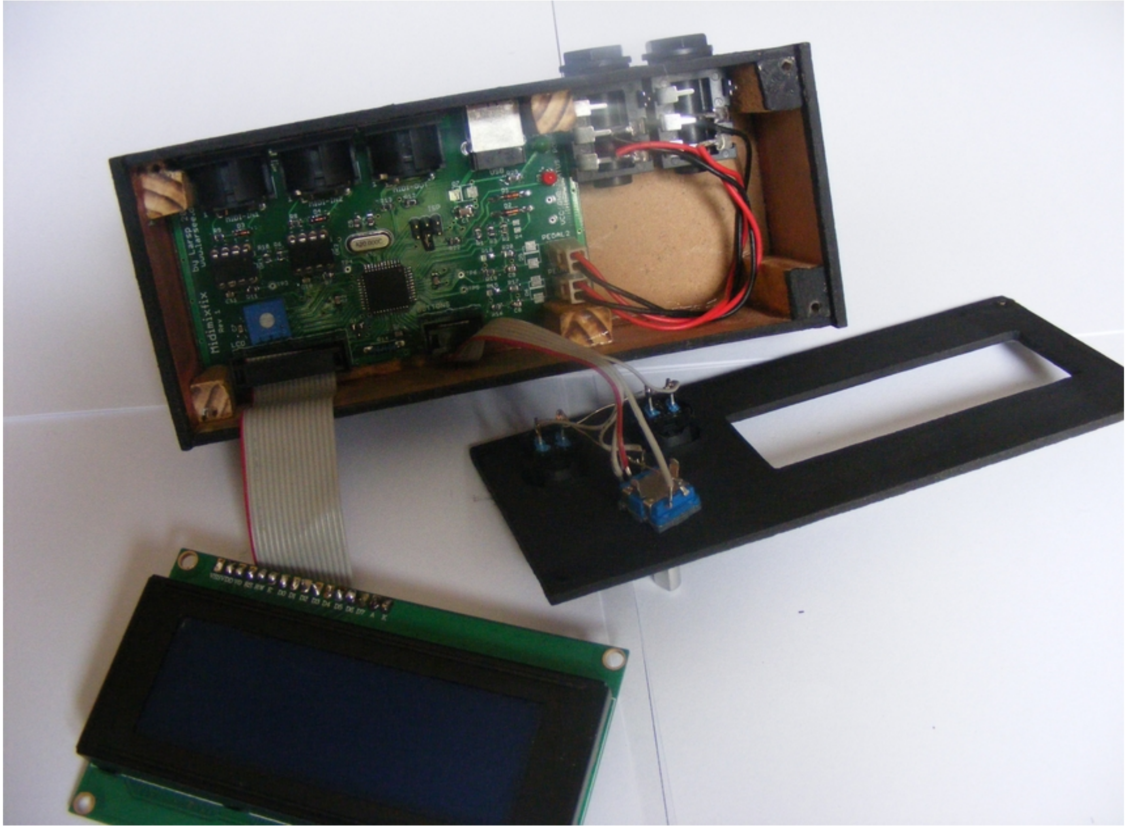

Fig. 7: Oh bugger! Since it is not a mass produced device the MIDIMIXFIX does not come with its own standardized housing. I therefore had to build my own little optimized enclosure.

11 The ultimate setup:

We are at last coming full circle now.
In the configuration described below below the R2M is combined with the GEM S3 into a setup that can turn any multitimbral MIDI synthesizer (my personal all time favorite being the Technics WSA1) into an instrument that has a controllability comparable to that of the Yamaha CS80.

Now here is our example as it actually appears in the MIDIMIXFIX's on board display:

Item:	Setting:	Specific remark no.:
Midi in 1	> 1	
Midi in 2	> 2	
Generate msg.	1 > 1	*1
When:	before N. on	
Type:	PitchW	
Val:	0	
Filtered route	2 > 1	*2
Type:	PitchW	*3
Curve 0-> 1	1 > 1	*4
On	N.on	
low:	1	*5
High:	128	*6
Curve:	Gamma 0.6	*7
Polyphon. Split	1 > 1	*8
Ch.range from:	9	*9
Ch.range to:	16	
Broadcast CC.:	yes	*10
Bcst. Progchg:	yes	
Bcst. P.wheel:	yes	
Bcst. Chan. AT:	yes	
MIDI out	> 1	
Runn. Status :	On	

General remarks:

- In this example The R2M is no longer used as a Trautonium type monophonic controller on itself but as a ribbon controller adding pitch bend data to the note data generated on the controller keyboard.
- The GEM S3´s MIDI out is therefore connected to the MIDIMIXFIX's MIDI IN 1 and the R2M to the MIDIMIXFIX's MIDI IN 2.
- The receiving multitimbral synth is connected to the MIDIMIXFIX´s MIDI OUT.
- The receiving synthesizer should obviously be set up to respond to notes on different MIDI channels. That, after all, is what multitimbrality is all aboutl. Since there are 16 MIDI channels in all the synth could thus provide data for 16 voices playing with aftertouch data generated per individual note. Few hardware synths ahve ever been able to do that. And still all channels will react to overall pitchbend information generated by the ribbon.
- As already remarked it is rather logical to have all channels on the receiving synth

drive the same sound. This is however not strictly neccesary. By using different sounds one can actually create very interesting randomized voice allocation effects, mimicking so called ¨hocketing¨ (= having a complete melody or progression being played by different instruments sounding one after each other).

- Also remember that there is no specific rule that says that side to side movement on a ribbon should always control pitch and poly aftertouch always control filtering / timbre per note. Basically a whole universe of modulation options becomes available. One could for instance also use the polyphonic aftertouch data to to generate pitch offsets and thus create individual vibrato per note.
- Do not forget that the pressure sensor on the ribbon can also still be allocated to anything one fancies.

Specific remarks:

*1
These 4 lines generate a pitchbend to 0 message for every note that is played on the keyboard. Thus remaining R2M pitchbend messages are automatically reset every time one generates a new note with the keyboard (see paragraph 8, point 1).

*2
These 2 lines make sure that the data from MIDI input 1 and thus from the R2M is mixed with that from input 1 and thus treated with the pitchbend messages generated AFTER the resets mentioned in the previous point.

*3
Very important in this setup is that the MIDIMIXFIX will only pass pitchbend data from the R2M on because of this setting. Otherwise hitting the R2M would alwys create an extra note command, it´s number depending on where you first touch the ribbon (see paragraph 8, point 3).

*4
These 5 lines determine how the data input form the keyboard is filtered.

*5
An important point specific to the MIDIMIXFIX is that the Low setting should normally never be set to 0. If one then plays the keyboard so softly that no aftertouch data is generated the whole note will be filtered from the output and thus no information about that note will ever reach your synth! Not even a Note On! This is actually a very neat feature to avoid MIDI overloads (Who for instance needs a note with a MIDI volume value of 0. Nothing will be heard anyway!) but if one is not aware of it this can actually become counterproductive (where are those darn notes I just hit!¨).

*6
The High setting can however be set at will. One can put in a lower value then 127 to avoid extreme values, for instance if the synthesizer reacts too strong to the variations in aftertouch data that are generated by the keyboard.
In Trautonium mode this is also the place to get rid of the annoyingly high fixed velocity that the R2M generates (see paragraph 8, point 2).The R2M will just keep sending its number 127 but by adjusting this value here the MIDIMIXFIX will change it's output into

anything you want. It's easy!
Basically setting both values closer to each other will lead to data compression. You will then lose full 127 step resolution but can thus avoid extreme values.

*7
There are many curves that can be used to further influence the data interpretation over the whole width of the keyboard, normally called keyboard scaling. I'll not discuss this further here because we then have to dive very deeply into o the MIDIMIXFIX's possibilities. This is just a curve I like when I use the GEM S3's keyboard to adjust data that will be put into the Technics WSA synths.)

*8
This section sets the range of the MIDI channels that the incoming poly aftertouch data will be split to, in this case channel 8 to 16.

*9
To keep the total amount of MIDI data generated a bit in check an 8 channel setup kjhas been chosen has been chosen (how very CS80 indeed) using channel 9 to 16. The receiving Technics WSA1 synthesizer is doing and admirable job but under critical circumstances some latency is encountered. So in general: The quiker your multitimbral instrument will respond to MIDI the better.

*10
These settings configure which data will eventually be filtered out to lessen the overall data output to the receiving synthesizer. If set to “yes” no data is suppressed. By selecting “no” one can suppress data if the need occurs. In an 8 channel setup as described here I do normally however not get latencies high enough to make such filtering necessary.

12 Still not enough?

The only thing that can top the setup described here will be to have a controller available which can generate multi-axis data per note. Enter those ultra-adaptive new controllers. Well, this setup CAN also polyphonicaly change multiple continuous MIDI control values at once (although there is of course only one axis to control).
Speaking in the terminology if such controllers: We have just added polyphonic Z-axis control PLUS a versatile monophonic y-axis contoller PLUS an extra monophonic Z-axis controller (the R2M´s own pressure sensitivity) to our run of the mill multitimbral synthesizer setup.
In other words: As far as conventional keyboard setups go the solutions described in this article are already quite sophisticated
Not bad at all!

13 For further reading / viewing:

- A detailed description of the Doepfer Trautonium Project can be fond here:

  http://www.doepfer.de/traut/traut_e.htm

- Hm. didn´t I already write an article about pimping the Doepfer R2M?

  http://www.brassee.com/writing.html#pimp

- So you want tot receive that PDF with my nifty ribbon controller musical scale design, eh? Well you are dead out of luck. But if you go to my website at

  www.brasee.com

  push the contact button on the main page and send me an email a miracle might still happen.

- A few videos about setting up the MIDIMIXFIX

  https://www.youtube.com/watch?v=JRt7MXFQuEk

  https://www.youtube.com/watch?v=yqUHe00-khM

  https://www.youtube.com/watch?v=R56Ov9PaZ4Q

M.W.E.M. Brassé
27 January 2017