Digital Extensions to Möller Opus 10263
Catonsville United Methodist Church
Catonsville, MD 21228

Theory of operation and maintenance manual

Contents

First Release:	2/11/2016
Revised:	6/8/2016
	11/15/2017

1.0 Overall Concept

The basic concept of a digital organ is shown in figure 1. The organ keys are connected to a digital keyboard encoder which generates electrical signals that a computer can recognize. The computer generates proper sounds from the keys that are pressed. That sound signal is amplified for use with a loudspeaker.

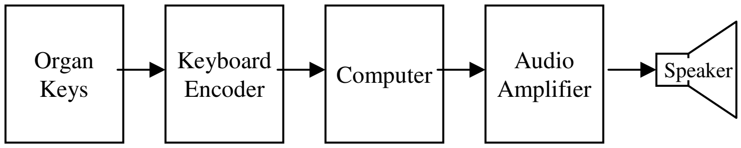


Figure 1.1. Digital organ concept

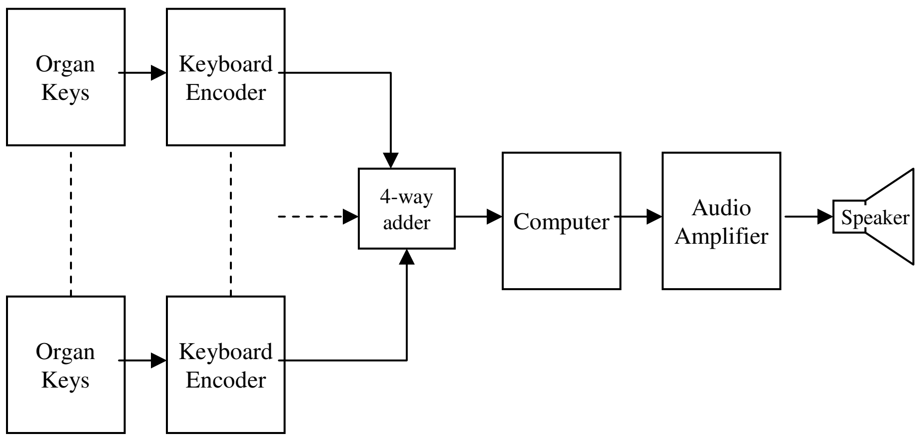


Figure 1.2. Expansion for multiple organ keyboards

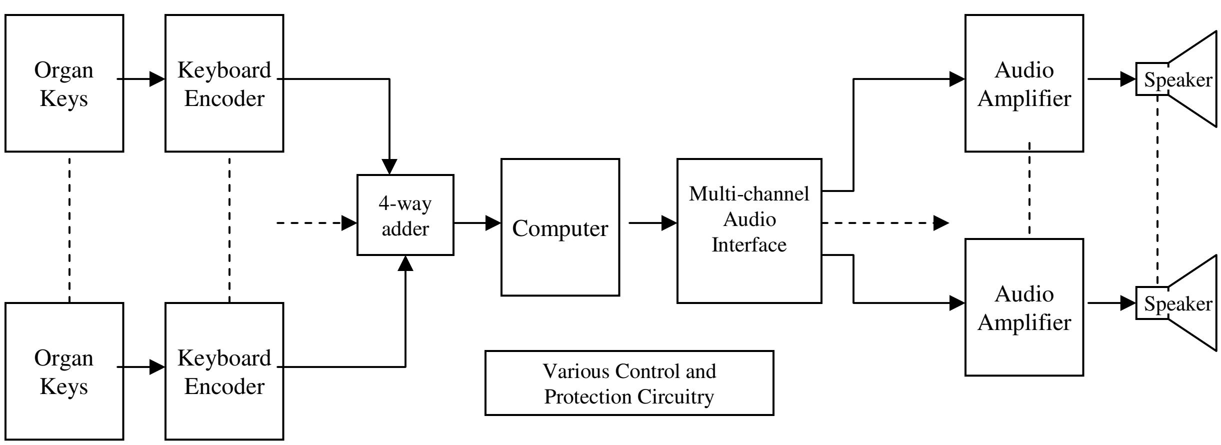


Figure 1.3. Expansion for multiple organ keyboards and organ divisions (Great, Swell etc.)

A complete block diagram of the digital installation at CUMC is shown in figure 1-4. This is an expansion of the basic concept from Figure 1-3 with various extensions that make the system practical when integrated with the existing Moller pipe organ. The function of the various assemblies need to be explained, as well as the locations and wiring points.

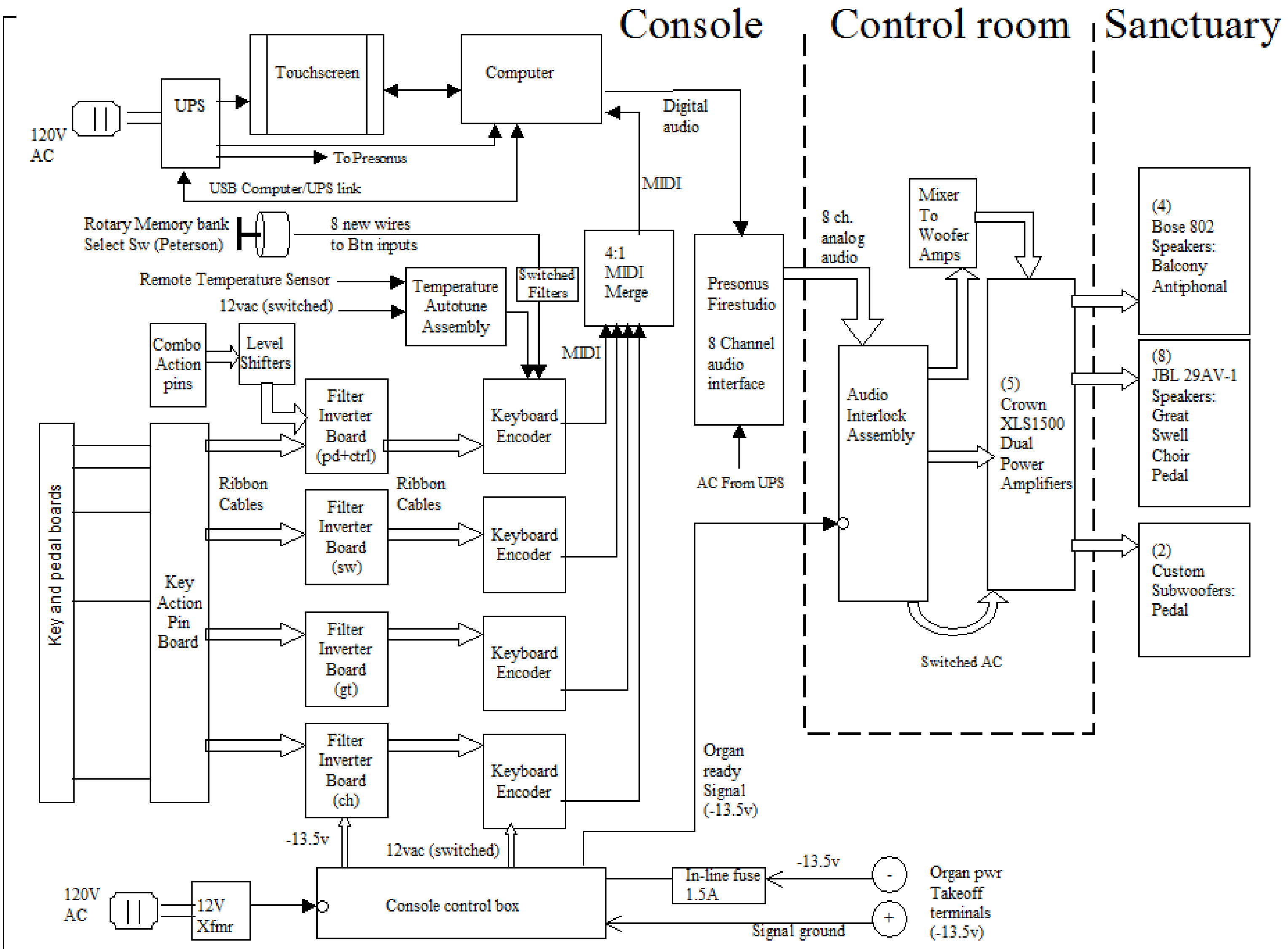


Figure 1.4. CUMC Digital Extensions Block Diagram

Component placement.

This is the organ console viewed from the rear with the back panel removed. The Filter-Inverter boards are located behind the keyboard encoders. This mounting board folds down for maintenance.

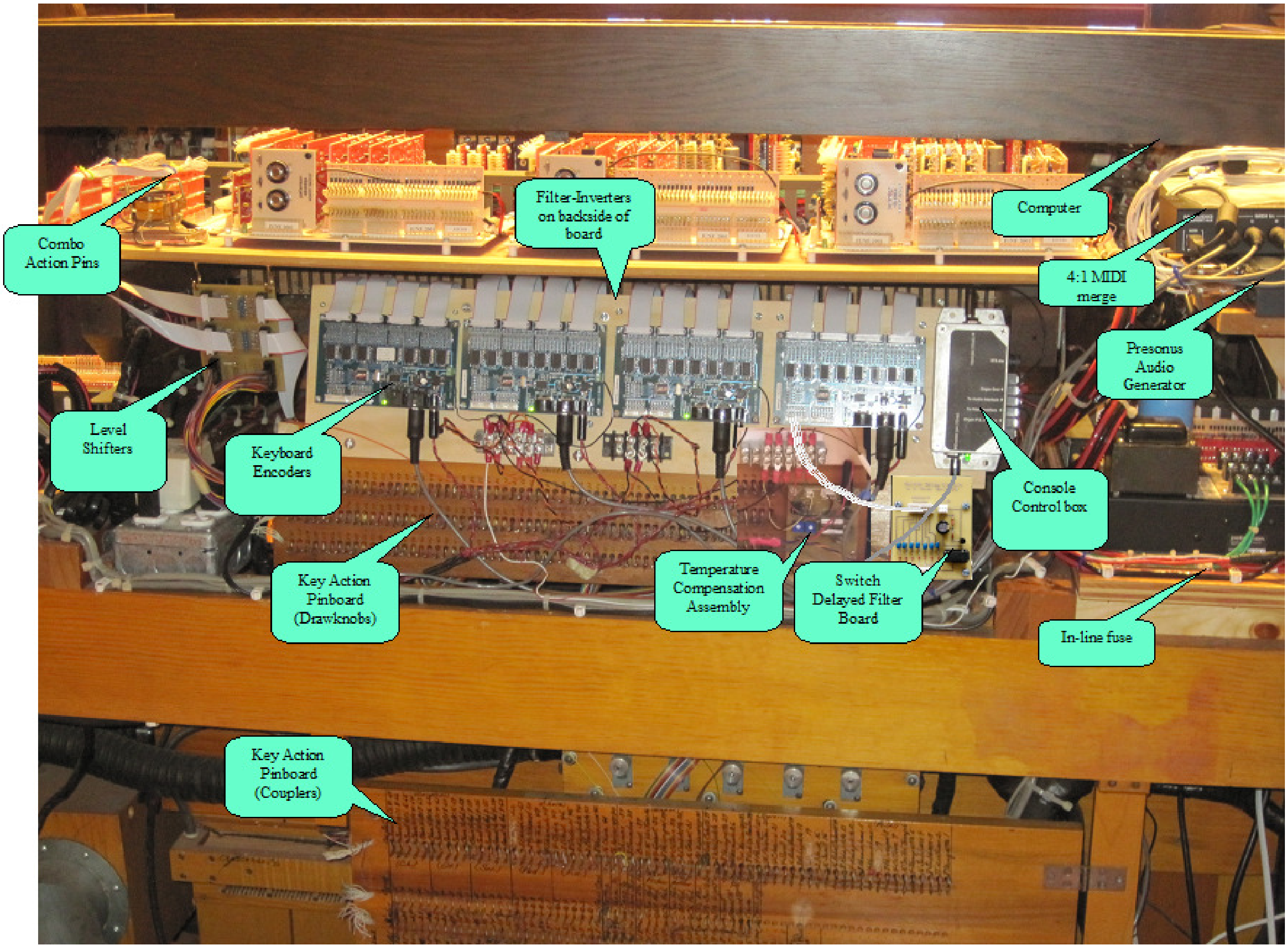


Figure 1.5. Component placement on the rear of the organ console.

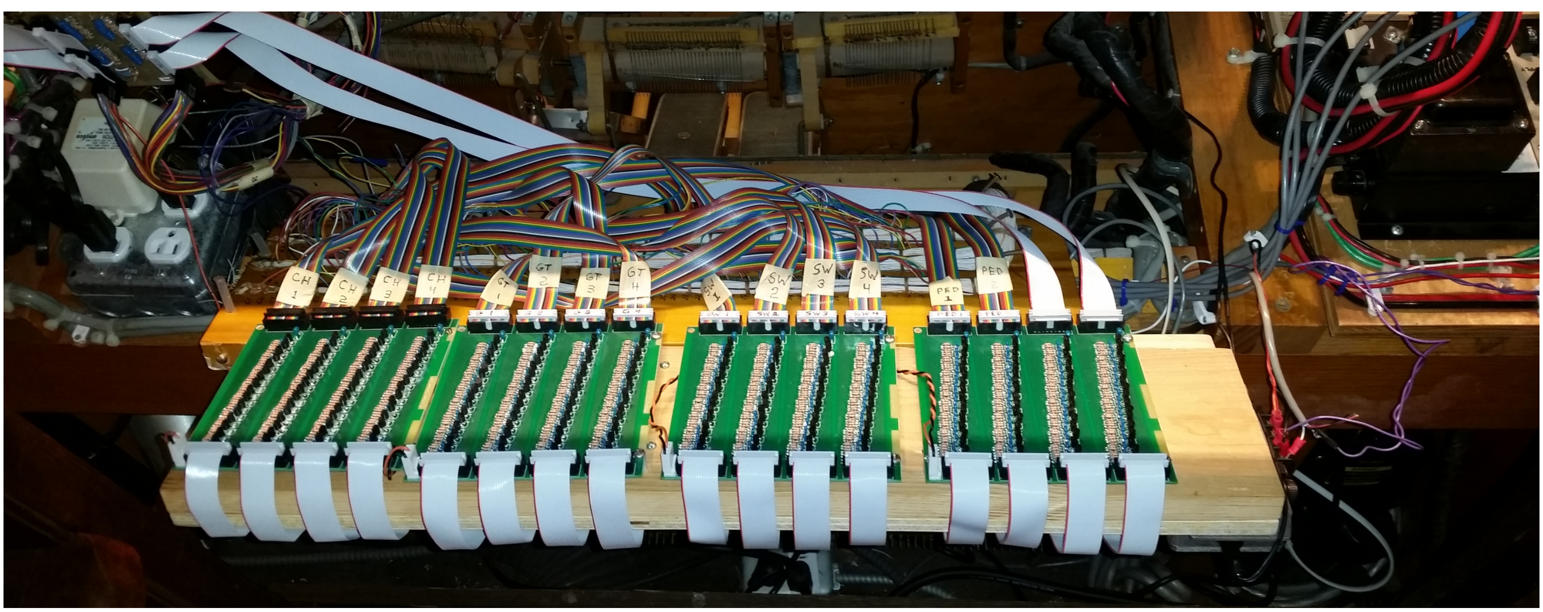
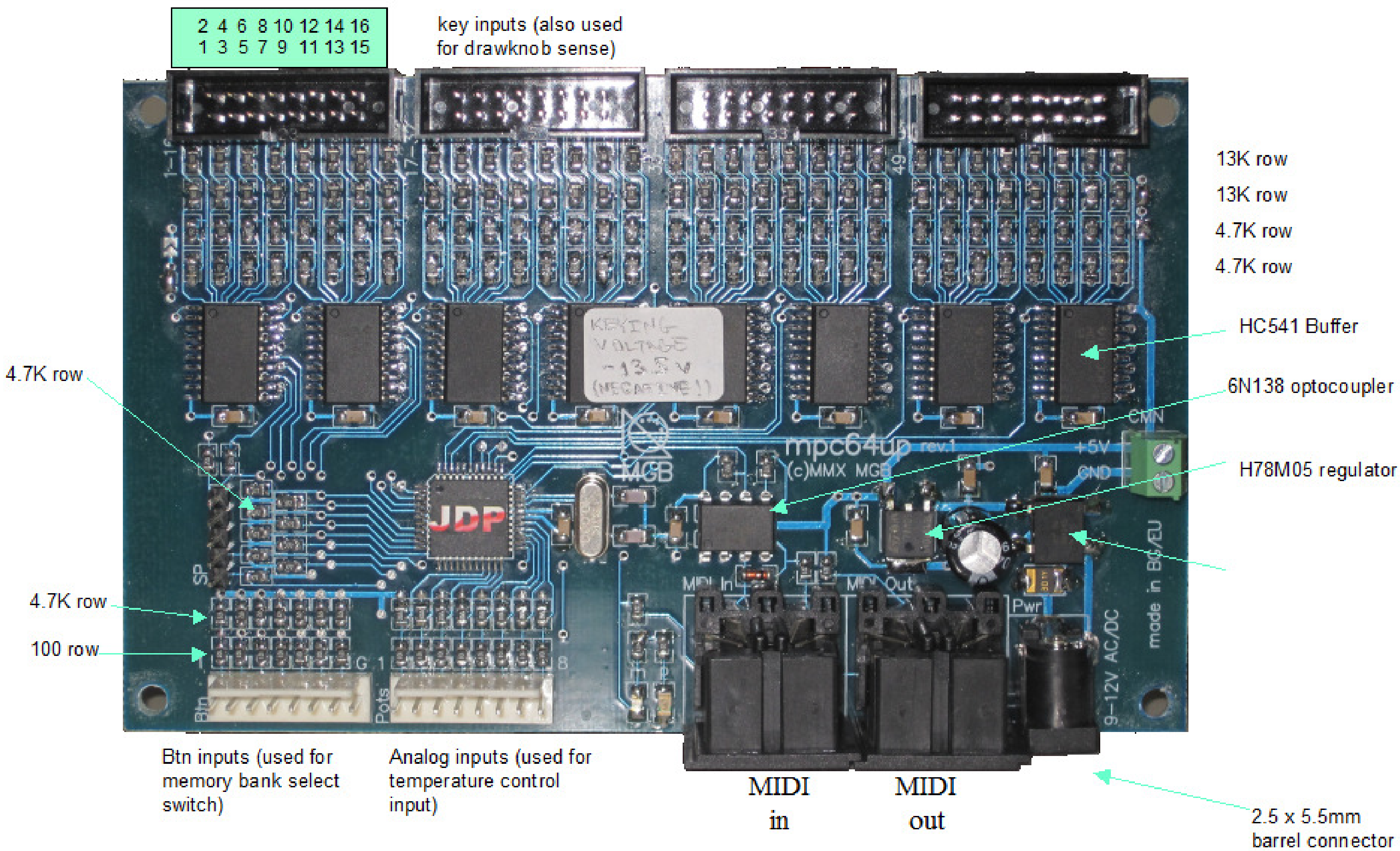


Figure 1.6. Filter Inverters as seen with mounting board folded down.

2.0 Component Descriptions

2.1 Keyboard Encoders

Four identical keyboard encoders are used for the three manuals and pedalboard. The encoders were bought from MidiGadgets Boutique, model mpc64up. The keying voltage was customized for the existing Moller installation which has a –13.5v "keyup" voltage and a 0v "keydown" voltage. This negative keying voltage is highly unusual and greatly complicated design of the digital extensions . However, that is the way the Moller was installed back in the 1960's and the digital extensions are designed to leave the original pipe organ operation unchanged.



Encoder sn001 has the 13K input resistors. Encoders 2,3,4 have 8.2K input resistors due to a misunderstanding when they were ordered. The mpc64up-usman.pdf user manual is include in the appendix.

2.2 Filter Inverters

The filter inverter boards are custom designed assemblies located between the organ keys and the keyboard encoders. These boards serve two functions. First, they act as low pass filters and attenuate switching noise generated by the Moller keying system. Short inductive spikes from the Moller keying system inductively couple between key circuits because there is up to 50 feet of cable bundling all the keying circuits to the pipe chambers. These inductive spikes were of no consequence to the original Moller action. The digital system responds much faster however and the short coupled spikes cause spurious short staccato notes.

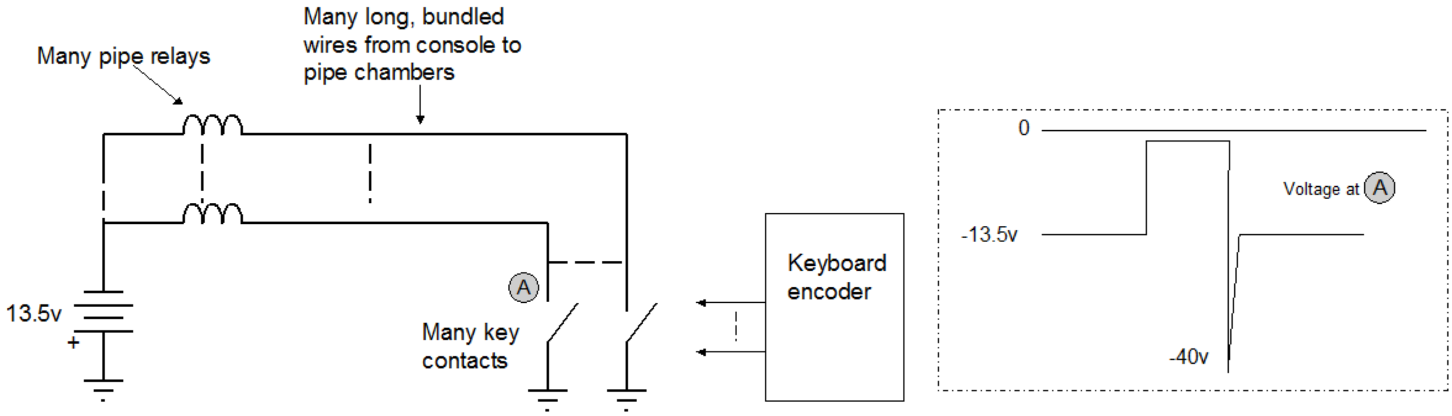


Figure 2.2-1 Keying a single pipe generates an inductive spike which couples to other circuits

The second function is to reverse the polarity of the keying voltage change. This required because of the very unfortunate negative polarity chosen by Moller when the organ was built. When the organ voltage is turned off for any reason, such as during shut-down, the keyboard encoders interpret this voltage change as key-down. The result would be a condition whereby all the digital keys would be down at once, with disastrous consequences. The inverter board reverses this condition to key-up, which is a safe condition.

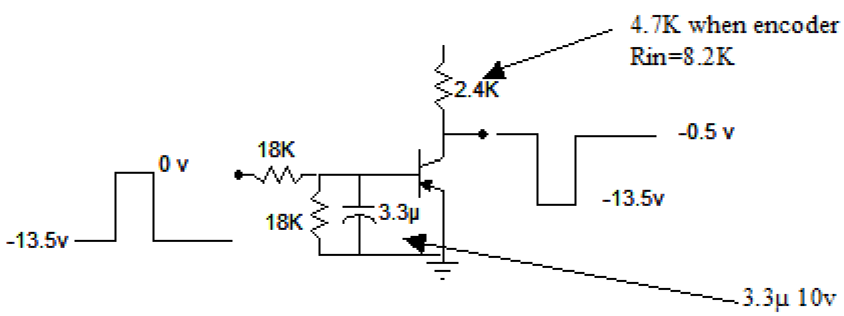


Figure 2.2-2 Filter-Inverter board circuit (typical)

Note that when an organ key is down (shorts to ground) the voltage is positive going at the input to the inverter board and negative going at the board output (keyboard encoder input). The 3.3uf capacitor slows the response such that keying transients are suppressed. This circuit is repeated 64 times for each keyboard encoder.

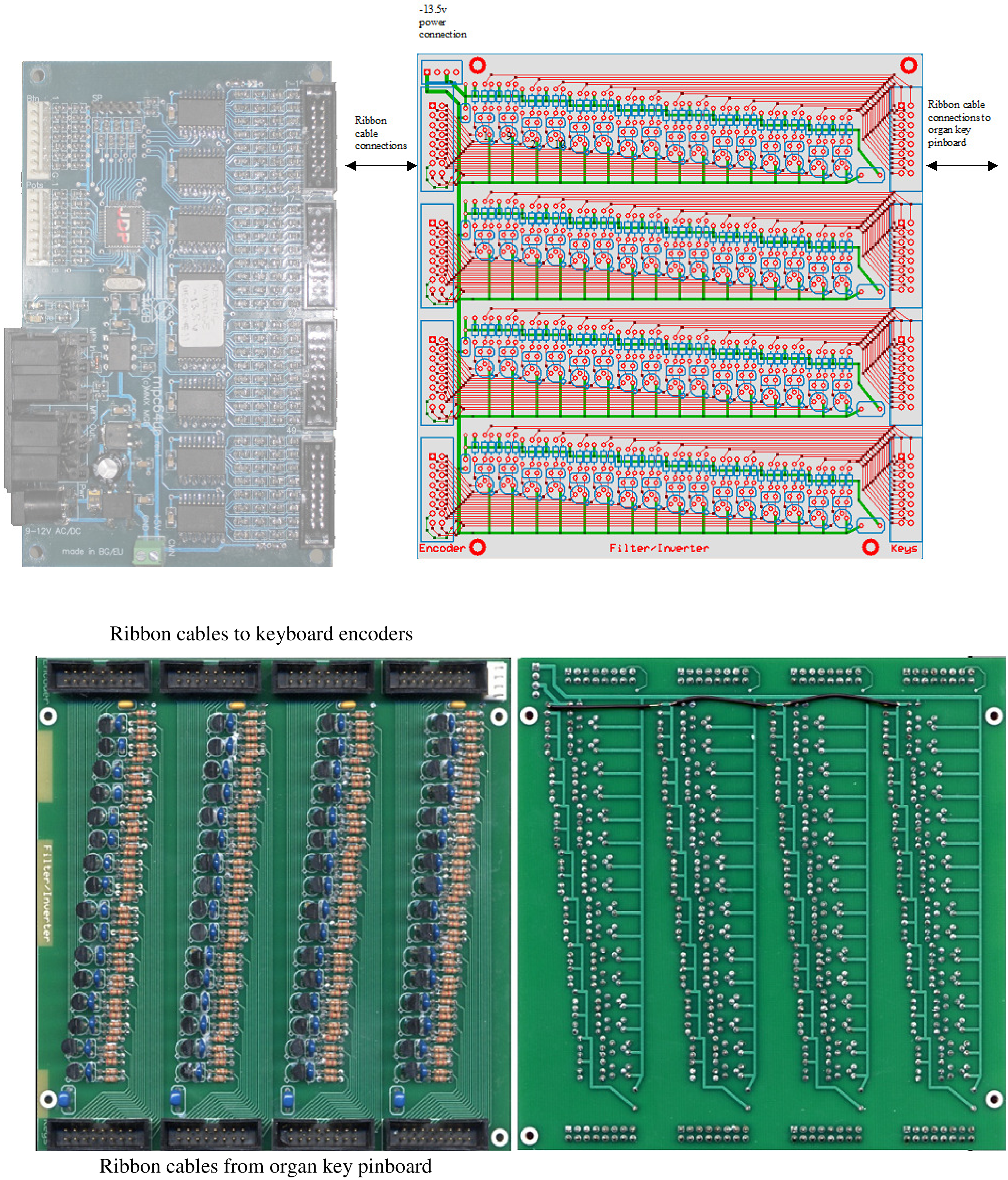
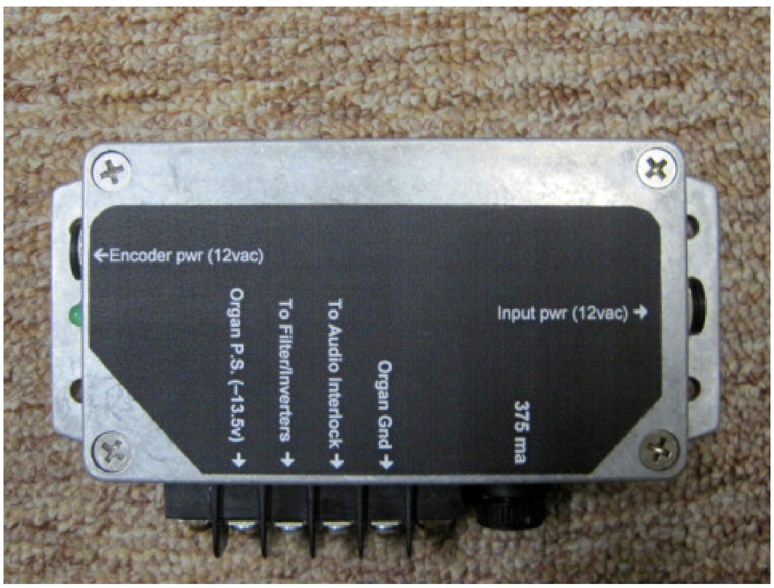


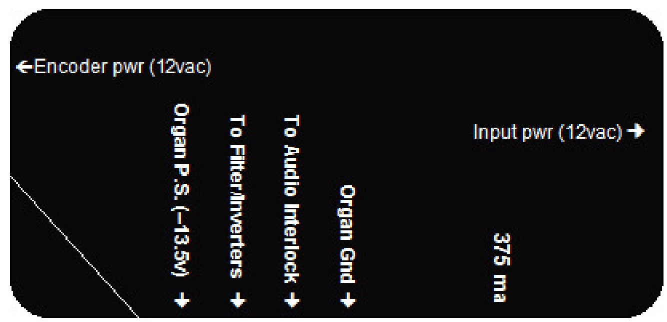
Figure 2.2-3 Physical layout and connection of the Filter-Inverter boards

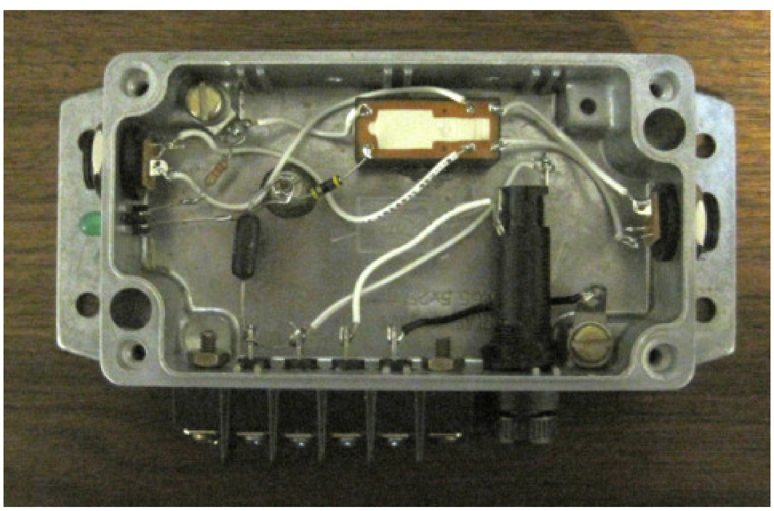
2.3 Console Control Box

The console control box accepts as input the main –13.5v Moller organ supply voltage. When this voltage is present, a relay is used to switch 12vac to the encoders from a 12vac wall wart power supply. The encoders will run off either 12vac or +12vdc, but not –12vdc. That is the reason a unique power supply is used instead of the console –13.5v power. The control box is also the tie point for voltage to the filter-inverters and the audio interlock unit in the control room.

The organ –13.5v supply voltage is routed through a 375ma fuse before being sent to the remote audio interlock assembly. If the remote audio interlock is not receiving an input Vsense voltage, then this fuse may be blown. The green LED indicates that the organ –13.5v voltage is available to the digital system. If it is not lit the most likely reasons would be that the inline fuse is blown or that the organ itself does not have it's dc voltage.







The input/output jacks are DigiKey CP-011B
These are 2.5mm ID, 5.5mm OD
This is the same size as the keyboard encoders.

The relay is DigiKey RT444012F-ND, 12V DPST
The coil is 360 ohm, 33.3 ma

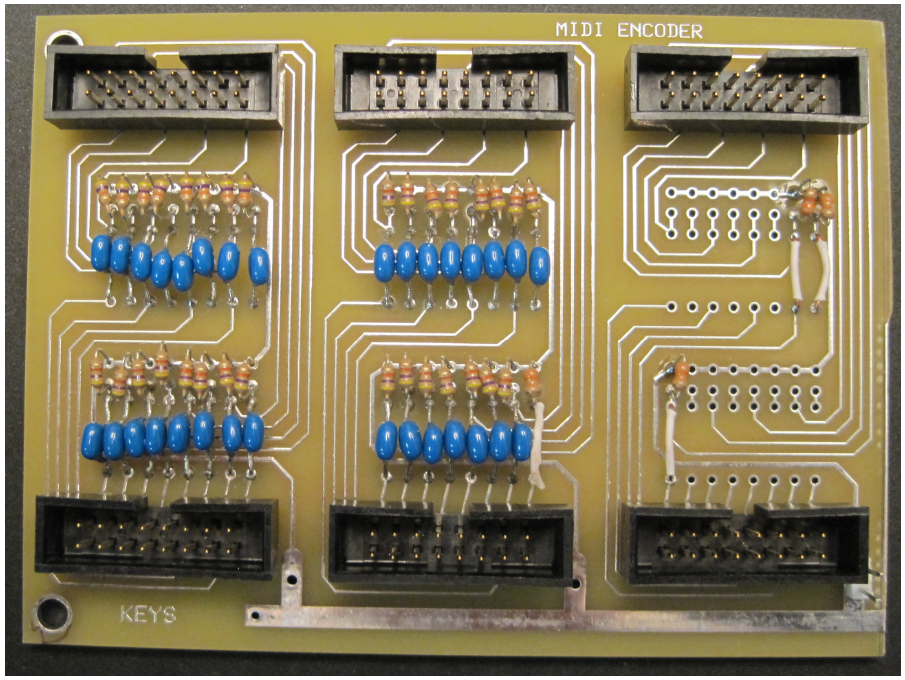
2.4 Level Shifter Assembly

The level shifter assembly converts the –5v signal from the Moller combination action pistons to a signal which is compatible with the Keyboard encoder/Filter-inverter assemblies. The output of this assembly is fed to the 32 inputs which would otherwise not be used in the 64-input Pedal board encoder. This assembly is hay-wired from a printed board which was originally intended as an RC filter for suppression of keying transients. It predates the present active Filter-Inverter boards.

Ribbon Cable connectors from the Combination action pins.

Level Shifter Assembly

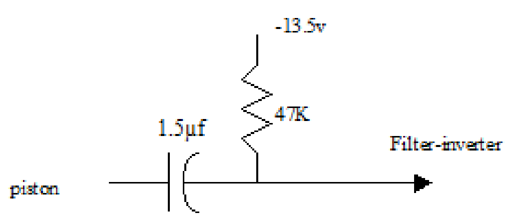
Ribbon Cable connectors to the Filter-Inverter board



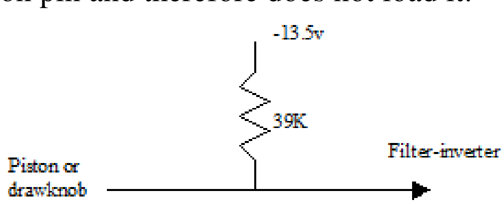
The Moller (Peterson) stop drawknobs all switch between nominal 0 and –13.6v, as do the Cancel and sFz buttons. The other combination action pistons all switch from –4.8v to 0 when pressed, as do the blank stop knobs.

The combination action pistons are AC coupled so that the signal used is momentary no matter how long the organist holds the piston. Three circuits are DC coupled since they need continuous action. They are the Setter and Cancel pistons and the drawknob which Moller had used for electronic chimes and is now used for the Great-Principal-16 stop.

The AC coupled piston circuit is:



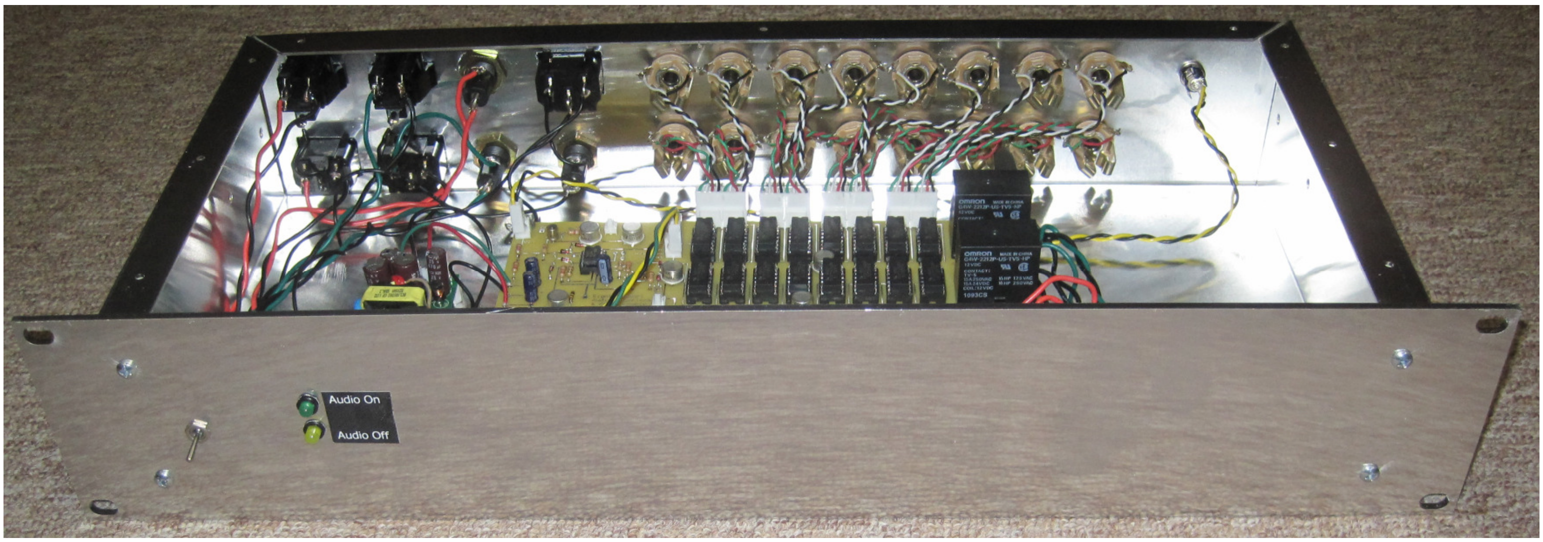
The direct coupled circuits contain just a pullup resistor to the organ –13.5v. The value is especially chosen such that the combination of the Level Shifter, Filter-inverter, and pullup resistor presents –4.8v to the Moller(Peterson) combination action pin and therefore does not load it:

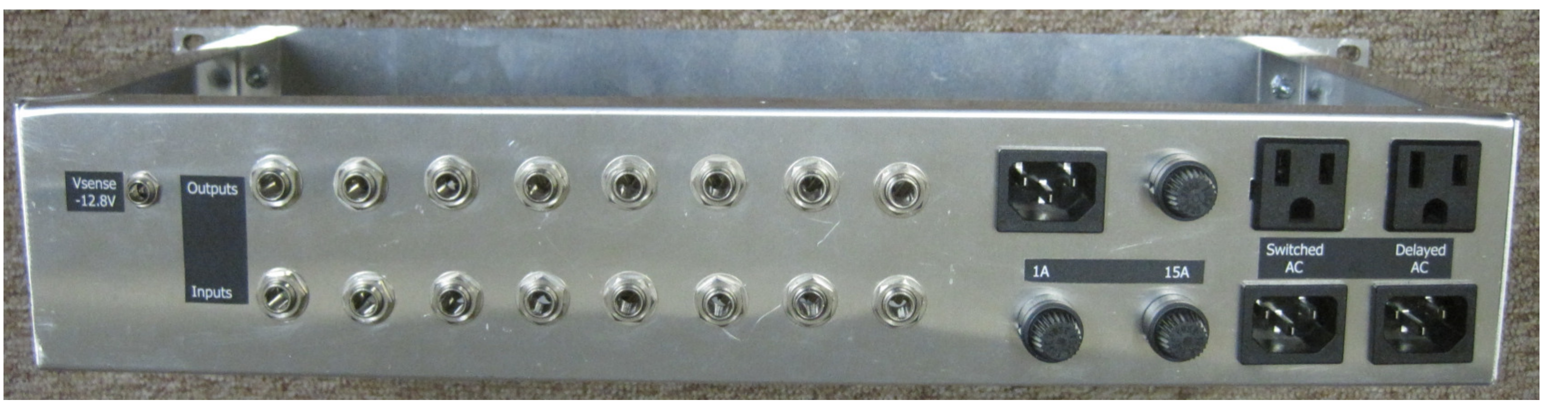


2.5 Audio Interlock Assembly

This a custom designed rack mounted assembly which interlocks and controls the audio power amplifiers. It controls eight balanced audio channels based on a nominal -12 volt enable signal which is sent from the organ console when the console voltages are properly active. The unit also provides two switched AC outlets for the power amplifiers, one of which is delayed by one second.

This interlock function is considered vital to safe operation of the digital organ system. The power amplifiers and speakers are very powerful and this is one layer of safety which aims to make sure spurious audio signals are not amplified during any turn-on, turn-off, or other unusual condition from the organ console and it's electronics. As previously discussed, the Filter-Inverter assemblies are another layer of safety which ensure that the digital keys and stops are "key-up" whenever the organ control voltages are not normal.





An LM293 comparator is used to sense the control voltage from the console. The interlock operates at threshold value of –10.5v and the nominal value sent from the console is the –13.5v Möller power supply voltage. A tripped condition requires a voltage level of -11.15v for reset. This hysteresis prevents chatter around the trip point. The 3.3µf capacitor gives the circuit some immunity from short noise spikes. The trip level is set by the resistor divider R6 and R7 which is compared to the 6.2v reference diode D6.

The audio relays are dual reed assemblies Digi-Key HE109-ND. One set of relay contacts shorts each audio input if the control voltage is not correct. A second set of relay contacts open the series path to the output connector. This is done to absolutely, positively disable the audio signals when the organ control voltage is not within limits. The power relays are Digi-Key Z3693-ND. The unit is powered by it's own internal ±15v dual power supply.

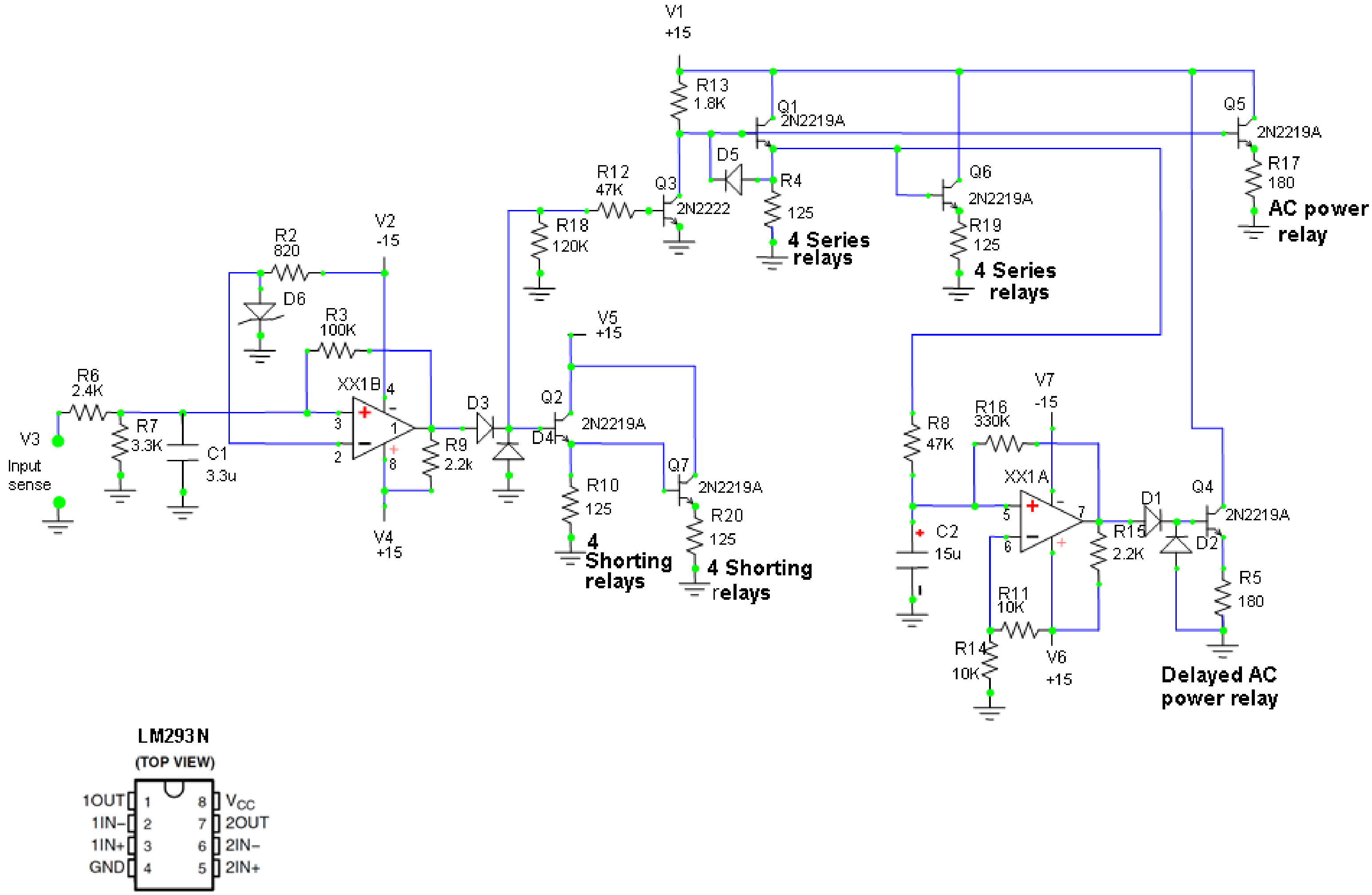


Figure 2.5-1 Schematic of the Audio Interlock Assembly

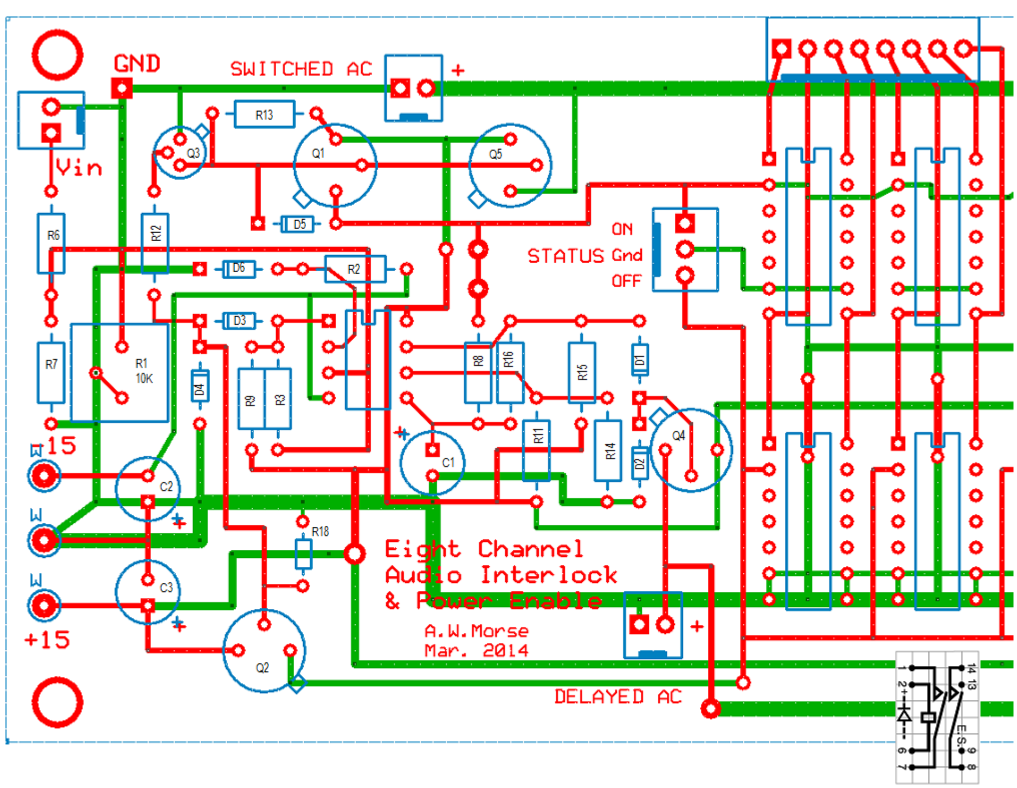
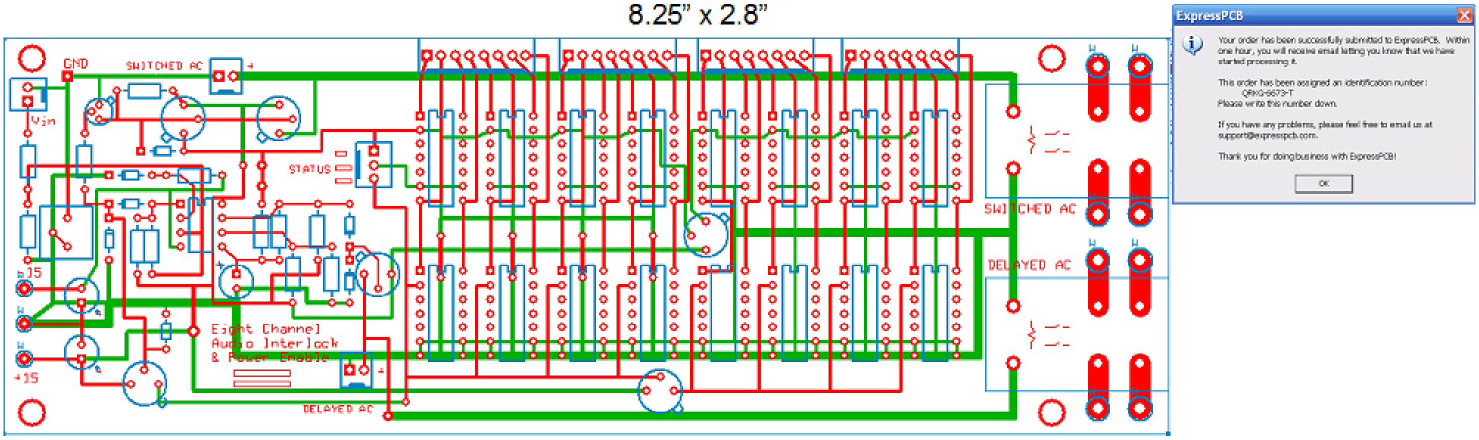
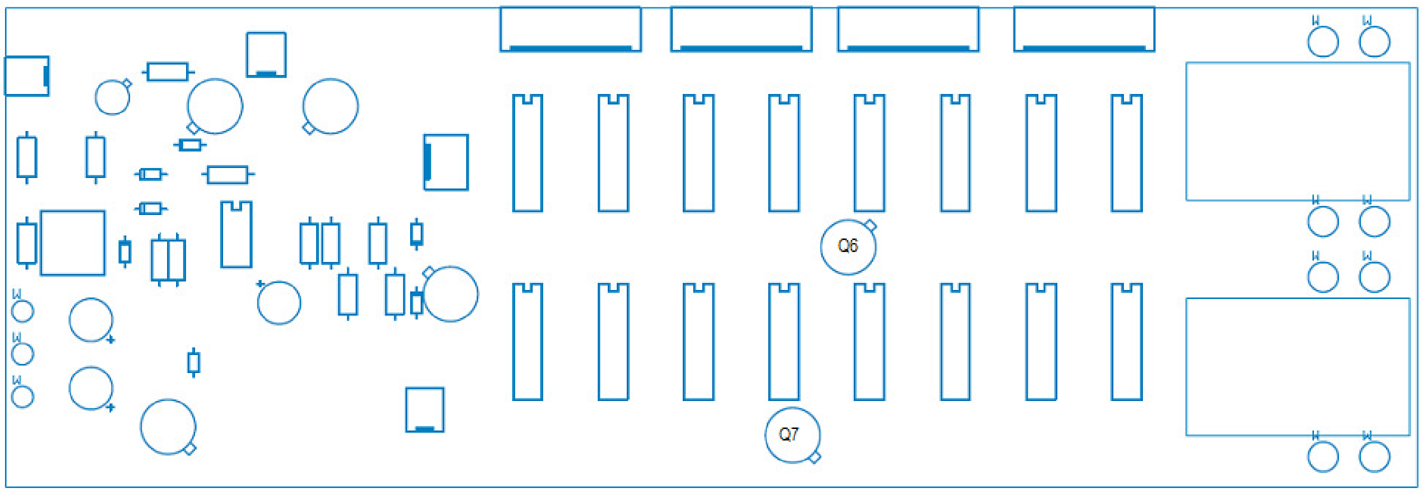


Figure 2.5-2 Component Identification of the Audio Interlock Assembly

8.25” x 2.8”





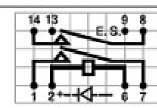
2A			HE722A0500	5	200	3.75	0.5	12	
DPST-NO	See Table 2 Column 1	Figure 1	HE722A1200	12	500	8.0	1.0	22	
			HE722A2400	24	2150	16.0	2.0	46	

Figure 2.5-3 Layout of the Audio Interlock Assembly

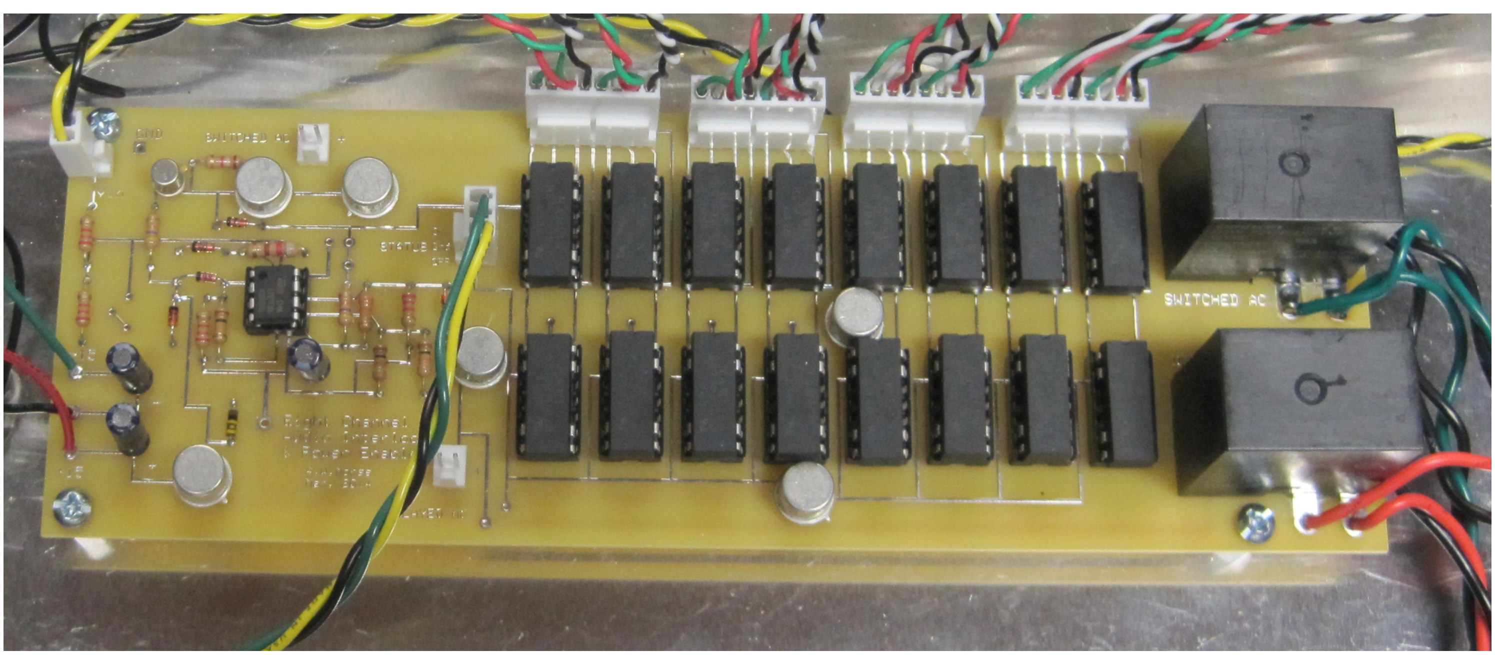


Figure 2.5-4 Circuit Board

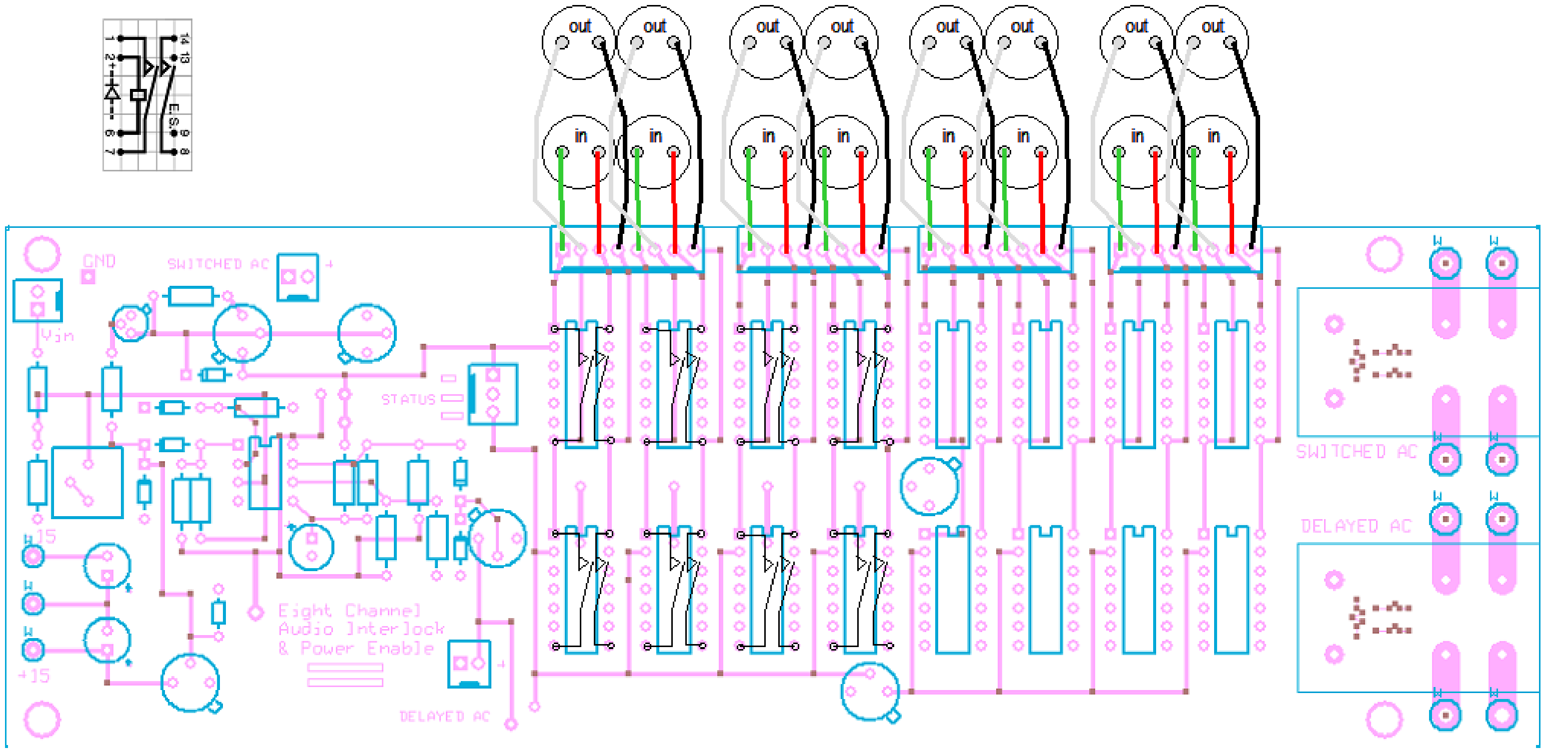


Figure 2.5-4 Wiring of the Balanced Audio Connectors

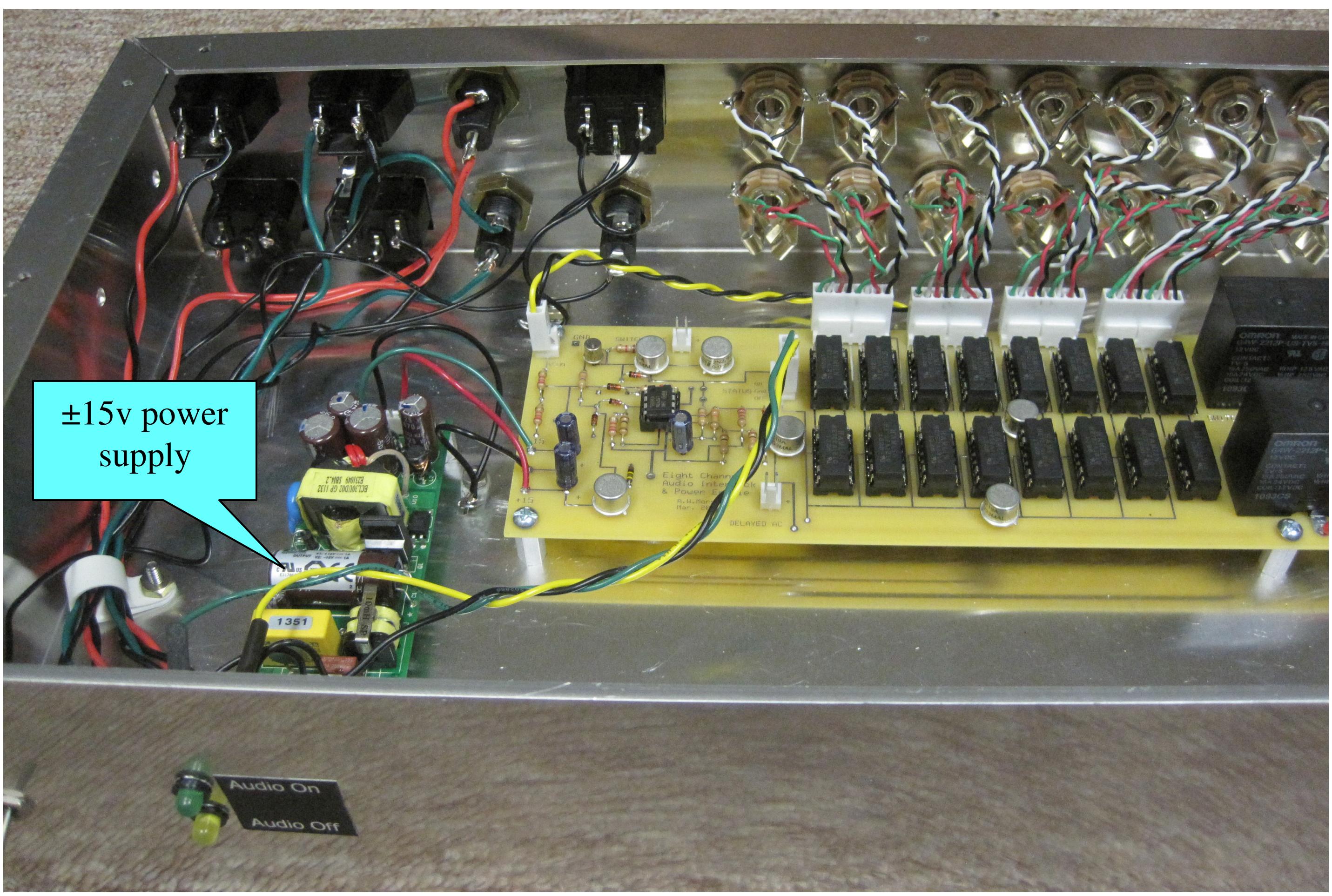


Figure 2.5-5 Internal Component wiring

2.6 Autotune Temperature Compensation

The pitch of the pipe organ changes due to variations in the pipe chamber air temperature. The digital ranks must track this change so that the entire hybrid organ remains in tune. The autotune circuitry senses the temperature in the pipe chamber and adjusts the digital pitch to match the pipes over a range of 52ºF to 92 ºF. The pipe organ is normally tuned at the center temperature of 72ºF.

As the air temperature changes from 52ºF to 92 ºF the velocity of sound varies from 338m/s to 351m/s. When you do the arithmetic, this translates to a pitch change of ±32 cents for a flue pipe. Reed pipes change much less and that means that a pipe organ does not stay perfectly in tune with itself as temperature changes. We correct all the digital ranks to track the pitch of the flue pipes.

The circuit that temperature tunes the digital ranks is shown in figure 2.6-1 and the actual circuit board in figure 2.6-2. The temperature sensor is an LM34. This device outputs a voltage of 10mv/ ºF . The circuit board contains a DC coupled amplifier which amplifies this to an output range from 0.3V to 4V. This voltage is fed to an analog input on the Pedal keyboard encoder, and organ software processes the resulting MIDI signal to adjust pitch.

The circuit provides two calibration voltages corresponding to 52ºF and 92 ºF . A slide switch is used to feed the full scale range represented by these calibration signals to the auto-detect function in the Hauptwerk software.

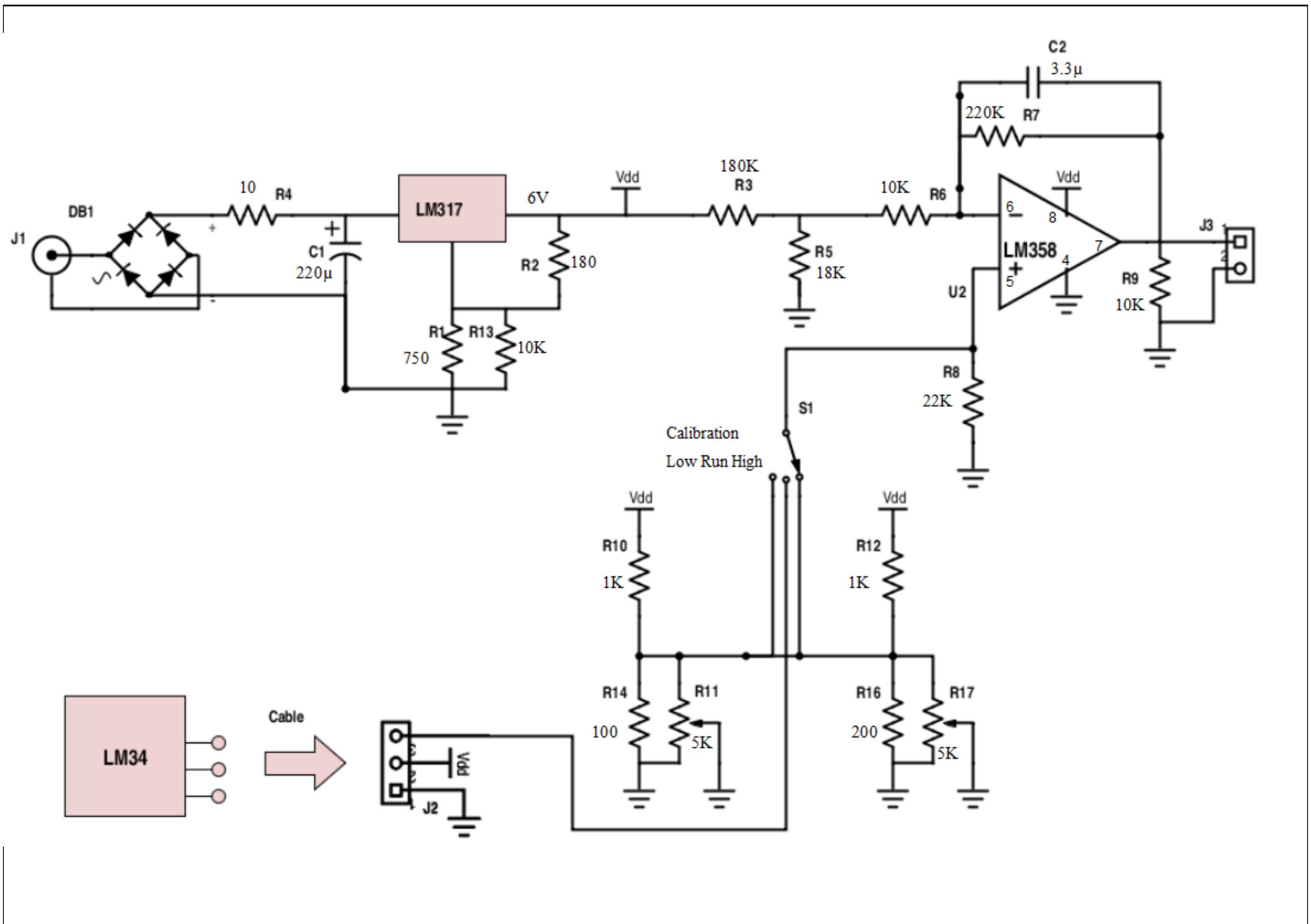


Figure 2.6-1 Autotune circuit

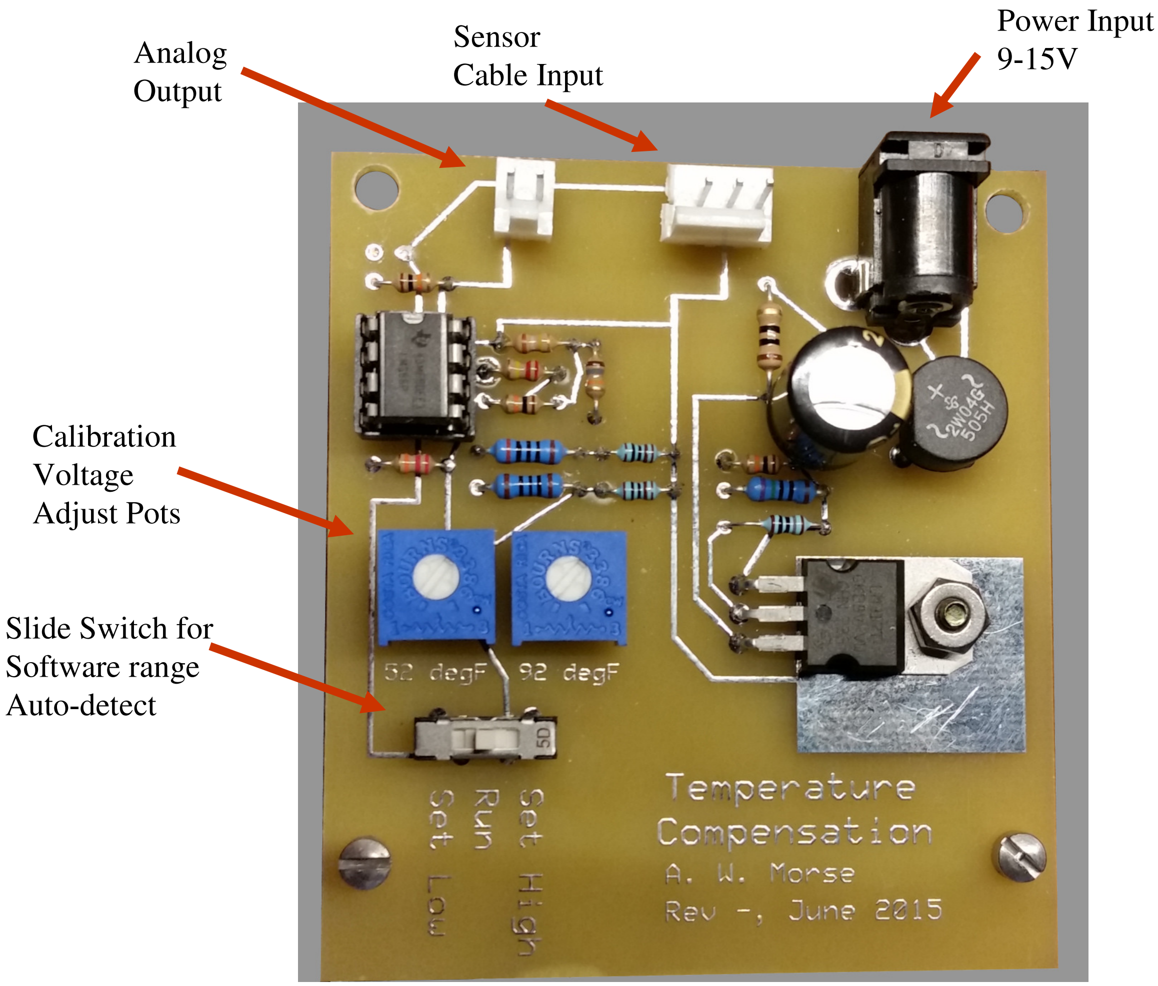


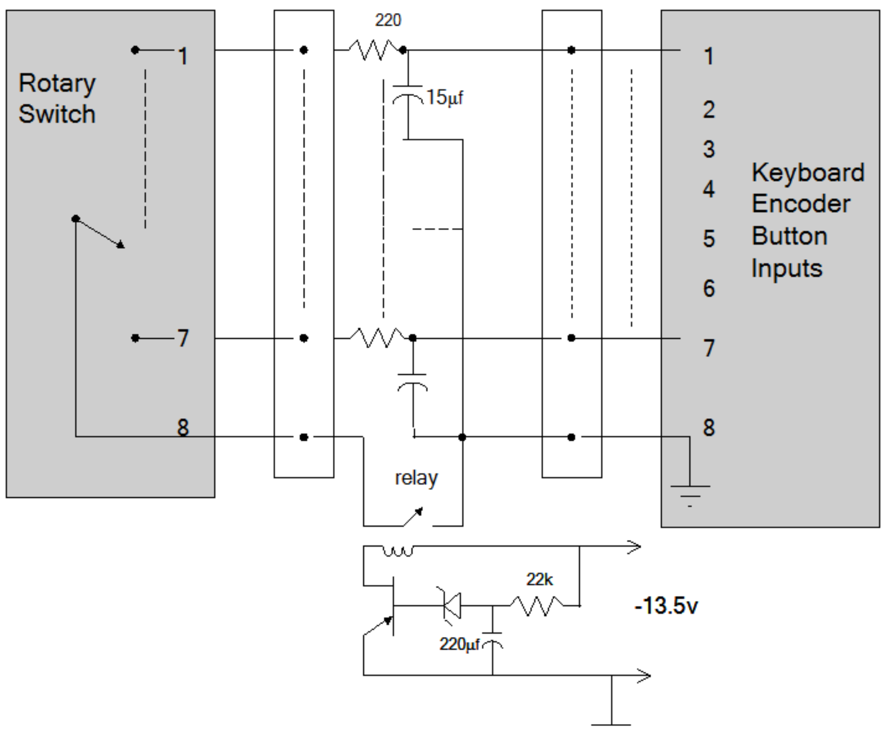
Figure 2.6-2 Autotune circuit board

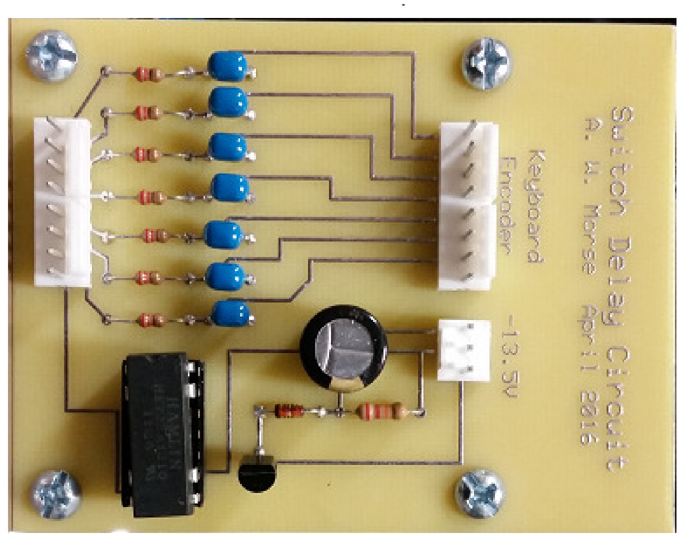
2.7 Switch Delayed Filter Board

The wires from the rotary switch which selects the digital registration bank are susceptible to emi pickup from the high leve pipe organ keying system. This can occasionally trigger the switch inputs of the digital keyboard encoder. These false triggers can cause the registration bank to switch without warning while the organ is played. This problem is solved by inserting RC filters between the rotary switch and the keyboard encoder inputs. The values used are 220 ohms and 15µf.

The switch inputs of the keyboard encoder have internal pull-up resistors to 5v. When the input is externally pulled to ground, the encoder outputs it's MIDI signal. Unfortunately, when the organ is first turned on, all the filter capacitors are discharged and the encoder interprets this as all the inputs activated, including the proper one as set by the rotary switch. The encoder reads all inputs sequentially before the capacitors have a chance to charge up and the Hauptwerk program quickly switches the registration bank eight times, ending at bank 7, the last Pedal encoder input. Hauptwerk always uses the last input it receives since there is no way to unselect a registration bank.

The solution to this is to delay the signal from the rotary switch until after the filter capacitors have had time to charge up. This is done by delaying the ground connection to the switch for 2 seconds after organ turn-on. The circuit is connected to the organ –13.5v supply and activated at the same time as the keyboard encoder. The DIP reed relay has a coil impedance of 500 ohms.





3.0 Sound System *(Design and Installation by Chuck Gehrman)*

3.1 Hauptwerk and Sound Production

The Hauptwerk program generates a digital audio stream which requires an audio interface to convert it to analog signals for amplification. The built-in audio of the Mac computer, while very good, is limited to two channels of audio out. Since Hauptwerk is capable of multiple output channels, and it was our intent to add voices to each division of the organ, the PreSonus Firestudio Project was chosen for our audio interface. This unit is moderately priced, and gives eight channels of output. Communication between the interface and the computer is by Firewire.

3.2 Audio Channels

In order to have the digital voices blend with the actual pipes, it was necessary to have the speakers for each division physically located in that division. This has the additional effect of letting the organ's swell boxes provide a realistic crescendo and decrescendo for all voices in the division. It was decided to use the eight channels of output as four stereo pairs. The pipe organ consists of four divisions, Swell, Great, Choir and Pedal. Since we were adding a fifth division, the Antiphonal, the digital pedal voices were split between the Swell and Great speakers to place them near similar voices. The 32' Pedal Contra Posaune speaks from and is expressive with the Swell. This puts it in the same chamber as the 16' Pedal Contra Trompette pipes. The remaining Pedal voices speak with the Great, which puts them outside of expression and on the right side of the instrument in the same area as the rest of the Pedal pipes.

3.3 Amplification and Speaker Selection

From the beginning of this project, it was our intention to add voices to the Moller organ that would be difficult or impossible to identify as non-winded. To this end, the decision was made to place most speakers inside the organ chambers so their output would mix normally with that of the pipes. The chamber placement meant that the directional characteristics of the speakers were not an issue. Horn speakers, which are efficient, but tend to beam their output, could be used since the chamber behaved as a mixing space. The decision was made to go with JBL Control 29-AV speakers as they combine relatively smooth frequency response with high efficiency and the ability (common to most speakers designed for PA use) to play quite loudly with little distortion. Once in place, the frequency response of each division was measured using white noise and a spectrum analyzer and appropriate correction applied to all samples intended for use in that division. The result was startlingly believable pipe sounds.

One stereo pair of speakers is used in each of the Choir and Great divisions. The Swell organ, which contains a number of powerful reed voices, including the 32' Pedal Contra Posaune, contains four speakers. To provide the copious amounts of clean power needed, it was decided to use Crown XLS 1500 amplifiers. These units have switching power supplies and are quite efficient and lightweight for their output. Each amp also has the ability to provide a loop-through output to mix for the subwoofers and a flexible crossover system with peak limiters built in.

In the Antiphonal Division, the speakers are four Bose 802 full range units. They are driven in stereo and equalized. Since they are not in a swell box, these voices are not expressive. They are mounted in boxes in the rear balcony openings. As such, they are the only clearly visible speakers in the organ, except for the subwoofers, which are not readily visible from the congregation.

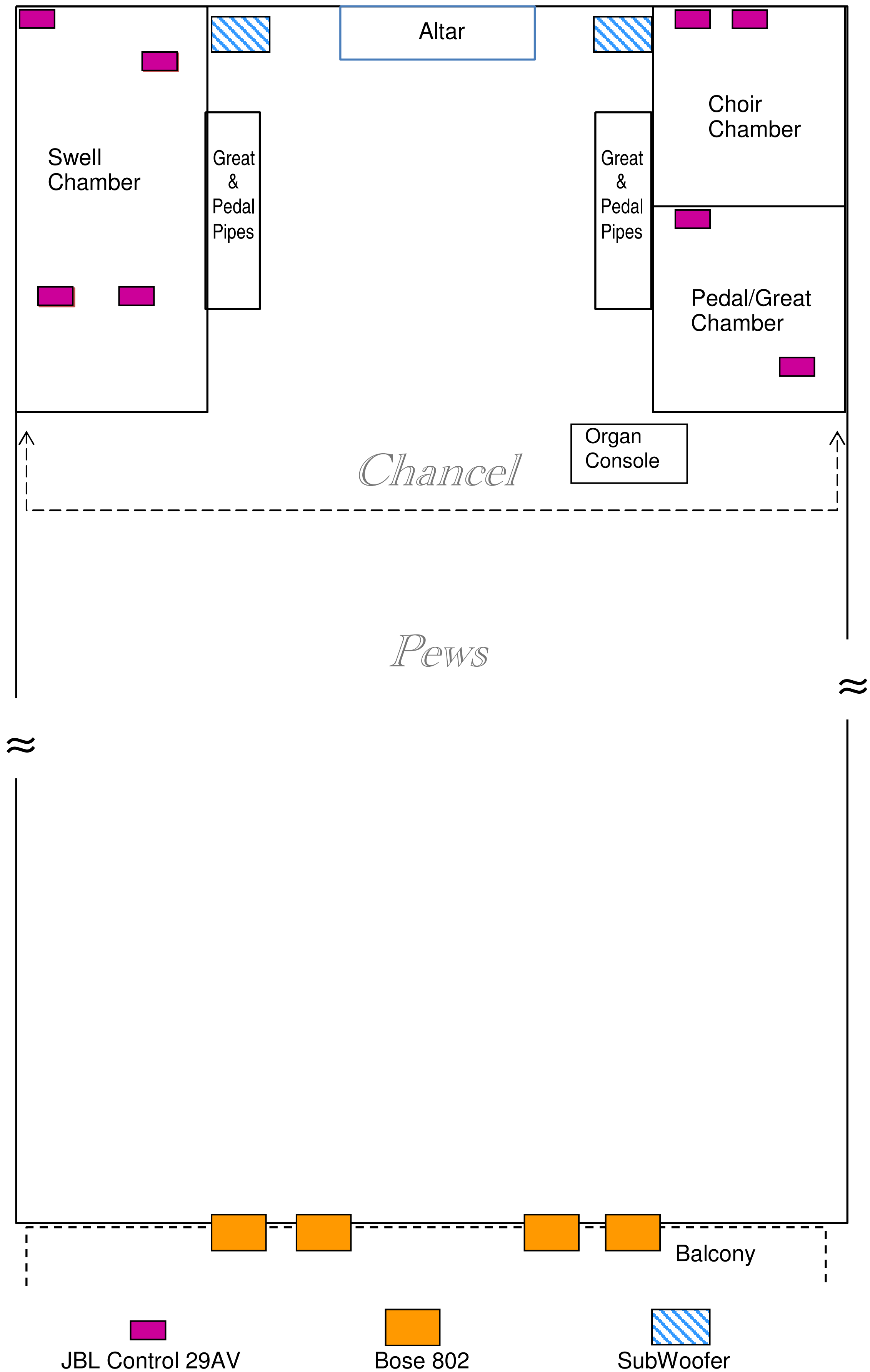


Figure 3.3. Speaker Placement

3.4 Routing of Channels and Audio Equipment Layout

The audio signals produced by the Presonus Firestudio located in the organ console are sent over an eight channel shielded and balanced snake to the Control Room beneath the Swell Organ chamber. The signals pass through the Audio Interlock and are distributed to four stereo amplifiers. The Choir Organ voices and the Antiphonal Organ voices are treated as full range signals and sent to the speakers in those locations. The Swell and Great voices include pitches below 8', including the various pedal stops. These amplifiers are set in Hi Pass mode, with the crossover frequency at 53 Hz. This sends all frequencies above 53 Hz to the speakers in the chambers. The unfiltered signal from each of these amplifiers is also passed-thru to the SubWoofer mixer. The signals from all four channels are summed and sent to the power amplifier for the subwoofers. This amplifier is set in Low Pass mode with the crossover frequency at 53 Hz. Thus all frequencies below 53 Hz are routed to the two SubWoofer cabinets. The physical layout of the equipment is shown in figure 3.4.1 below.

Audio Interlock

2 Power Distribution Units

Sub Woofer Amplifier

Great Amplifier

Sub Woofer Mixer

Antiphonal Amplifier

Choir Amplifier

Swell Amplifier

Figure 3.4. Audio Equipment Rack

4.0 Software

4.1 Hauptwerk Program Installation

Digital extensions to the organ are manipulated through the software program "Hauptwerk" from Milan Digital Audio. The Advanced version of the program is used and the church's license is contained in the USB "dongle" installed in the computer. Detailed operation of the program is always available in the installation manual available from Milan Digital Audio. Hauptwerk version 4.2.1 is used and it would be unwise to ever update it. Some of the custom programming (temperature control and rotary switch reads) depends on specific Hauptwerk protocols which could be changed by future whims of Milan Digital Audio.

The organ sample set normally loaded is named "Moller 3m103-autotune". The "Stops" screen page contains all the touch controls that are required for normal use. This screen does not show the four additional stops that are assigned to physical drawknobs which were available on the console. These stops are the Gt: Principal 16', Gt: Trompete 8', Sw: Tuba Mirabilis 8', and the Ch: Erzahler Celeste 8".

The two tremulant controls on the touch screen are normally activated automatically when the "Tremulant" drawknobs are pulled on the console. The touch screen controls can be used override this setting, however.

The large "C" button can be used to quickly cancel just the digital stops and does not affect the physical drawknobs. The indicator in the lower left corner displays the current air temperature and the correction which has been applied to the digital stops by the autotune circuitry. This can be tweaked by the "Fine Tune" slider at the top of the screen, but this is not normally necessary.

Catonsville United Methodist Church
M. P. Möller
Opus 10263
1967 (2015 Digital)

GREAT ORGAN

Principal	16'	
Principal	8’	61 pipes
Bourdon	8’	61 pipes
DoppelFlöte	8’	
Octave	4’	61 pipes
RohrFlöte	4’	
Quint	2 2/3'	
Doublette	2’	61 pipes
Sesquialtera	II	
Mixture	II-IV	208 pipes
Trompete	8'	
Tromba	8'	

SWELL ORGAN

Rohrgedekt	16’	12 pipes
HohlFlöte	8'	
Rohrflöte	8’	61 pipes
Viole de Gambe	8’	61 pipes
Viole Celeste	8’	49 pipes
Gamba Celeste II	8’	
Principal	4’	61 pipes
Flute	4’	
Waldflöte	2’	61 pipes
Plein Jeu	III	183 pipes
Mixture IV	IV	
Basson	16’	
Trompette	8’	61 pipes
Tuba Mirabilis	8’	
Oboe	8’	
Vox Humana	8’	
Clarion	4’	12 pipes
Oboe	4’	
Tremulant		

CHOIR ORGAN

Gedeckt	8’	61 pipes
Erzahler	8’	61 pipes
Erzahler Celeste	8’	
Flute Celeste II	8’	
Salicional	8’	
Principal	8’	
Prestant	4’	
Koppelflöte	4’	61 pipes
Nazard	2-2/3’	61 pipes
Blockflöte	2’	61 pipes
Terz	1-3/5’	61 pipes
Krummhorn	8’	61 pipes
Clarinet	8’	
English Horn	8’	
Zymbelstern		
Tremulant		

PEDAL ORGAN

Resultant	32’	
Contra Posaune	32’	
Untersatz	32’	
Contra Basse	16’	
EchoBass	16’	
Open Diapason	16’	
Subbass	16’	32 pipes
Rohrgedeckt (Sw)	16’	
Contre Trompette	16’	12 pipes
Principal	8’	32 pipes
Subbass	8’	12 pipes
Rohrflöte(Sw)	8’	
Octave	4’	12 pipes
Rohrflöte(Sw)	4’	
Mixture	II	64 pipes
Trompette (Sw)	8’	
Krummhorn (Ch)	8’	
Trompette (Sw)	4’	
Krummhorn (Ch)	4’	

ANTIPHONAL

Principal	8’
Octave	4’
Super Octave	2'
Tuba Imperial	8’
Trompette Chamade	8’
Harmonic Trumpet	8’
Chimes	
Carillon	

COUPLERS

Great to Pedal	Swell to Pedal
Great to Pedal 4'	Swell to Pedal 4'
Choir to Pedal	
Choir to Pedal 4'	
Swell to Great 16'	Choir to Great 16'
Swell to Great	Choir to Great
Swell to Great 4'	Choir to Great 4'
Swell to Choir 16'	
Swell to Choir	
Swell to Choir 4'	
Great 16'	Swell 16'
Great Unison Off	Swell Unison Off
Great 4'	Swell 4'
Choir 16'	Antiphonal to Swell
Choir Unison Off	Antiphonal to Great
Choir 4'	Antiphonal to Choir

4.2 Sample Set Programming

Only one sample set is normally used with this installation and it has been customized to augment the Möller opus 10236 and the Peterson control system that was installed in 2001. The ODF (Organ Definition File) is based on a standard CODM compilation. The only unique programming is that which is necessary to add the automatic temperature compensation.

The pitch of any pipe organ varies significantly with the air temperature. The digital electronics are constant with temperature and the two thus become out of tune with each other when the temperature varies from where the organ was tuned. The Hauptwerk software allows the digital pitch to be manually adjusted to match the pipes over temperature. It is inconvenient (and embarrassing) to adjust the digital tuning when the congregation is present, especially if the temperature varies during a service. For this reason, it was worthwhile to incorporate an autotuning feature in the digital system.

Figure 4.2.1 shows the effect of temperature on the speed of sound in air. When the temperature rises, the speed of sound increases and the effective length of a pipe (in wavelengths) increases. This causes the pitch to go flat. For an organ tuned at 72° F, the pitch changes 32% of a semitone (cents) for a temperature change of 20° Fahrenheit. The autotune system is programmed to correct for a temperature range of 72±20 °F which corresponds to a pitch range of ±32 cents.

Temperature of air ϑ in °C	Speed of sound c in m/s
+40	354.94
+35	351.96
+30	349.08
+25	346.18
+20	343.26
+15	340.31
+10	337.33
+5	334.33
0	331.30
−5	328.24
−10	325.16
−15	322.04
−20	318.89
−25	315.72

Figure 4.2.1 Effect of temperature on sound

The autotune circuit board produces a voltage which is proportional to temperature. This analog voltage can be read into the software system through any analog MIDI input – just as is commonly done with expression pedals. The MIDI information is then used to adjust the pitch of the audio samples so that they track the temperature shift of the organ pipes. Standard CODM does not contain the proper statements for this task and it necessary to alter the XML ODF file itself.

For each pipe defined in in “Pipe_SoundEngine01_Layer” the line for the pitch offset is incremented by 32 cents, the maximum amount that the pitch is to be raised:

```
<PitchLvl_DetuningPercentSemitones>32</PitchLvl_DetuningPercentSemitones>
```

and the line calling out a continuous control for pitch is altered:

```
<PitchLvl_ScalingContinuousControlID>261</PitchLvl_ScalingContinuousControlID>
```

This is done with a simple XML editor such as “XML Marker”. The required changes are illustrated using the parsing tool output of XML Marker on the ODF.

Two Continuous Controls are defined and added to the ODF:

Tag name/Text	Text
ControlID	260
Name	TUNE
DefaultInputOutputContinuousCtrlAsgnCode	260
AccessibleForInput	Y
AccessibleForOutput	Y
DefaultValue	63
RememberStateFromLastLoad	N
Clickable	N
ClickingHigherIncreasesValue	N
ImageSetInstanceID	200

Tag name/Text	Text
ControlID	261
Name	TUNEScaled
DefaultInputOutputContinuousCtrlAsgnCode	261
AccessibleForInput	N
AccessibleForOutput	Y
DefaultValue	63
RememberStateFromLastLoad	N
Clickable	N
ClickingHigherIncreasesValue	N
ImageSetInstanceID	

ControlID 260 is the visible indicator for the the touchscreen display and is also the control that will be connected to the analog MIDI input port. ControlID 261 is the control called out in the <Pipe_SoundEngine01_Layer> for actual pitch control. The default value for the controls is the midrange value of 63, since controls have a 0 to 127 range and the midrange value must lower the default 72° pitch from the maximum 32 cent offset which was set in the <PitchLvl_DetuningPercentSemitones> statement.

Tag name/Text	Text
ImageSetID	140
Name	Autotune
InstallationPackageID	800626
ImageWidthPixels	100
ImageHeightPixels	44
ClickableAreaLeftRelativeXPosPixels	0
ClickableAreaRightRelativeXPosPixels	69
ClickableAreaTopRelativeYPosPixels	0
ClickableAreaBottomRelativeYPosPixels	69
TransparencyMaskBitmapFilename	

The image set for the input control and indicator.

There are 41 images in Image Set 140. This is number 16, F67.jpg

Tag name/Text	Text
ImageSetInstanceID	200
Name	Tindicate
ImageSetID	140
DefaultImageIndexWithinSet	21
DisplayPageID	2
ScreenLayerNumber	12
LeftXPosPixels	0
TopYPosPixels	770

The image instance for the input control and indicator.

There are 41 stages to the output indicator, 52° to 92°, in 1° steps. The SetStage and SetElement tags are set thusly:

Tag name/Text	ImageSetID	HighestContinuousControlValue	ImageSetIndex
ContinuousControlImageSetStage	140	3	1
ContinuousControlImageSetStage	140	6	2
ContinuousControlImageSetStage	140	9	3
ContinuousControlImageSetStage	140	12	4
ContinuousControlImageSetStage	140	15	5
ContinuousControlImageSetStage	140	19	6
ContinuousControlImageSetStage	140	22	7
ContinuousControlImageSetStage	140	25	8
ContinuousControlImageSetStage	140	28	9
ContinuousControlImageSetStage	140	31	10
ContinuousControlImageSetStage	140	34	11
ContinuousControlImageSetStage	140	37	12
ContinuousControlImageSetStage	140	40	13
ContinuousControlImageSetStage	140	43	14
ContinuousControlImageSetStage	140	46	15
ContinuousControlImageSetStage	140	50	16
ContinuousControlImageSetStage	140	53	17
ContinuousControlImageSetStage	140	56	18
ContinuousControlImageSetStage	140	59	19
ContinuousControlImageSetStage	140	62	20
ContinuousControlImageSetStage	140	65	21
ContinuousControlImageSetStage	140	68	22
ContinuousControlImageSetStage	140	71	23
ContinuousControlImageSetStage	140	74	24
ContinuousControlImageSetStage	140	77	25
ContinuousControlImageSetStage	140	81	26
ContinuousControlImageSetStage	140	84	27
ContinuousControlImageSetStage	140	87	28
ContinuousControlImageSetStage	140	90	29
ContinuousControlImageSetStage	140	93	30
ContinuousControlImageSetStage	140	96	31
ContinuousControlImageSetStage	140	99	32
ContinuousControlImageSetStage	140	102	33
ContinuousControlImageSetStage	140	105	34
ContinuousControlImageSetStage	140	108	35
ContinuousControlImageSetStage	140	112	36
ContinuousControlImageSetStage	140	115	37
ContinuousControlImageSetStage	140	118	38
ContinuousControlImageSetStage	140	121	39
ContinuousControlImageSetStage	140	124	40
ContinuousControlImageSetStage	140	127	41

Tag name/Text	ImageSetID	ImageIndexWithinSet	Name	BitmapFilename
ImageSetElement	140	1	AutoTuneStage01	Images/AutoTune/F52.jpg
ImageSetElement	140	2	AutoTuneStage02	Images/AutoTune/F53.jpg
ImageSetElement	140	3	AutoTuneStage03	Images/AutoTune/F54.jpg
ImageSetElement	140	4	AutoTuneStage04	Images/AutoTune/F55.jpg
ImageSetElement	140	5	AutoTuneStage05	Images/AutoTune/F56.jpg
ImageSetElement	140	6	AutoTuneStage06	Images/AutoTune/F57.jpg
ImageSetElement	140	7	AutoTuneStage07	Images/AutoTune/F58.jpg
ImageSetElement	140	8	AutoTuneStage08	Images/AutoTune/F59.jpg
ImageSetElement	140	9	AutoTuneStage09	Images/AutoTune/F60.jpg
ImageSetElement	140	10	AutoTuneStage10	Images/AutoTune/F61.jpg
ImageSetElement	140	11	AutoTuneStage11	Images/AutoTune/F62.jpg
ImageSetElement	140	12	AutoTuneStage12	Images/AutoTune/F63.jpg
ImageSetElement	140	13	AutoTuneStage13	Images/AutoTune/F64.jpg
ImageSetElement	140	14	AutoTuneStage14	Images/AutoTune/F65.jpg
ImageSetElement	140	15	AutoTuneStage15	Images/AutoTune/F66.jpg
ImageSetElement	140	16	AutoTuneStage16	Images/AutoTune/F67.jpg
ImageSetElement	140	17	AutoTuneStage17	Images/AutoTune/F68.jpg
ImageSetElement	140	18	AutoTuneStage18	Images/AutoTune/F69.jpg
ImageSetElement	140	19	AutoTuneStage19	Images/AutoTune/F70.jpg
ImageSetElement	140	20	AutoTuneStage20	Images/AutoTune/F71.jpg
ImageSetElement	140	21	AutoTuneStage21	Images/AutoTune/F72.jpg
ImageSetElement	140	22	AutoTuneStage22	Images/AutoTune/F73.jpg
ImageSetElement	140	23	AutoTuneStage23	Images/AutoTune/F74.jpg
ImageSetElement	140	24	AutoTuneStage24	Images/AutoTune/F75.jpg
ImageSetElement	140	25	AutoTuneStage25	Images/AutoTune/F76.jpg
ImageSetElement	140	26	AutoTuneStage26	Images/AutoTune/F77.jpg
ImageSetElement	140	27	AutoTuneStage27	Images/AutoTune/F78.jpg
ImageSetElement	140	28	AutoTuneStage28	Images/AutoTune/F79.jpg
ImageSetElement	140	29	AutoTuneStage29	Images/AutoTune/F80.jpg
ImageSetElement	140	30	AutoTuneStage30	Images/AutoTune/F81.jpg
ImageSetElement	140	31	AutoTuneStage31	Images/AutoTune/F82.jpg
ImageSetElement	140	32	AutoTuneStage32	Images/AutoTune/F83.jpg
ImageSetElement	140	33	AutoTuneStage33	Images/AutoTune/F84.jpg
ImageSetElement	140	34	AutoTuneStage34	Images/AutoTune/F85.jpg
ImageSetElement	140	35	AutoTuneStage35	Images/AutoTune/F86.jpg
ImageSetElement	140	36	AutoTuneStage36	Images/AutoTune/F87.jpg
ImageSetElement	140	37	AutoTuneStage37	Images/AutoTune/F88.jpg
ImageSetElement	140	38	AutoTuneStage38	Images/AutoTune/F89.jpg
ImageSetElement	140	39	AutoTuneStage39	Images/AutoTune/F90.jpg
ImageSetElement	140	40	AutoTuneStage40	Images/AutoTune/F91.jpg
ImageSetElement	140	41	AutoTuneStage41	Images/AutoTune/F92.jpg

When a <PitchLvl_ScalingContinuousControl> goes over it's full range, the pitch changes from the starting value all the way to zero, which is of course too much. So the linkage from input Control 260 to action Control 261 must be scaled for the much smaller desired range. This is done thus:

ContinuousControlLinkage

(This tag has no attibutes.)

14 Subtags:

Tag name/Text	Text
SourceControlID	260
DestControlID	261
Name	TuneScaled
LinkTypeCode	1
WillisTypeIncSpeedInMillisecondsPerStepWithOneStepDiff	0
InertiaModelTypePositiveAcceleratingCoeff	
InertiaModelTypePositiveDampingCoeff	
ConditionSwitchID	
ConditionSwitchLinkIfEngaged	N
ReevaluateIfCondSwitchChangesState	Y
InvertSourceControlValue	N
SourceControlValueIncrement	3338
SourceControlValueCoefficient	.036661
SourceControlValueIndex	1

The values for the <ContinuousControlLinkage> which does the scaling may be calculated thus:

ContinuousControlLinkage

(This tag has no attibutes.)

14 Subtags:

Tag name/Text	Text
SourceControlID	260
DestControlID	261
Name	TuneScaled
LinkTypeCode	1
WillisTypeIncSpeedInMillisecondsPerStepWithOneStepDiff	0
InertiaModelTypePositiveAcceleratingCoeff	
InertiaModelTypePositiveDampingCoeff	
ConditionSwitchID	
ConditionSwitchLinkIfEngaged	N
ReevaluateIfCondSwitchChangesState	Y
InvertSourceControlValue	N
SourceControlValueIncrement	a
SourceControlValueCoefficient	b
SourceControlValueIndex	1

$$F = fo * \frac{((ctrl+a)*b)}{127}$$

Where "ctrl" is the value of the continuous control and goes from 0 to 127

For the frequency to vary over a 64 cents range (±32 cents) the multiplying factor $\frac{((ctrl+a)*b)}{127}$ must go from 0.963339 to 1.0 as the continuous control setting changes from 0 to 127.

The formula can be solved by simple algebra to give a=3337.154 and b=0.36661, which are the values inserted in the scaling linkage.

4.3 Sample Preparation and Voicing

All the samples used are "dry"with the intent that the digital sounds will mimic the exact acoustics of the "dry" actual pipes. The samples are played through speakers which are next to the real pipes of their respective divisions. There are no multiple releases to program, and since the Swell and Choir speakers are in the actual expression boxes, there are no artificial expression chambers to simulate.

The loudspeaker placement was diagramed in figure 3.3.1. It can be seen that litle direct sound enters the seating area of the sanctuary from the Great, Swell or Choir speakers. Most sound reaching the listener arrives after first bouncing off multiple walls and objects. This is also the case for most pipes since only the Great and Pedal Principals are exposed on open chests which are cantilevered above the chancel. Accordingly, the perceived placement of the "digital pipes" in each division is identical to that of the real pipes.

Each division uses between two and four speakers. The multiple reflections reaching a listener, from the multiple speakers and multiple paths, results in very good mixing with no noticeable comb filtering. There is, however, a very strong effect on the the overall transmission frequency curve, and very strong compensation was applied to the rank samples before amplification.

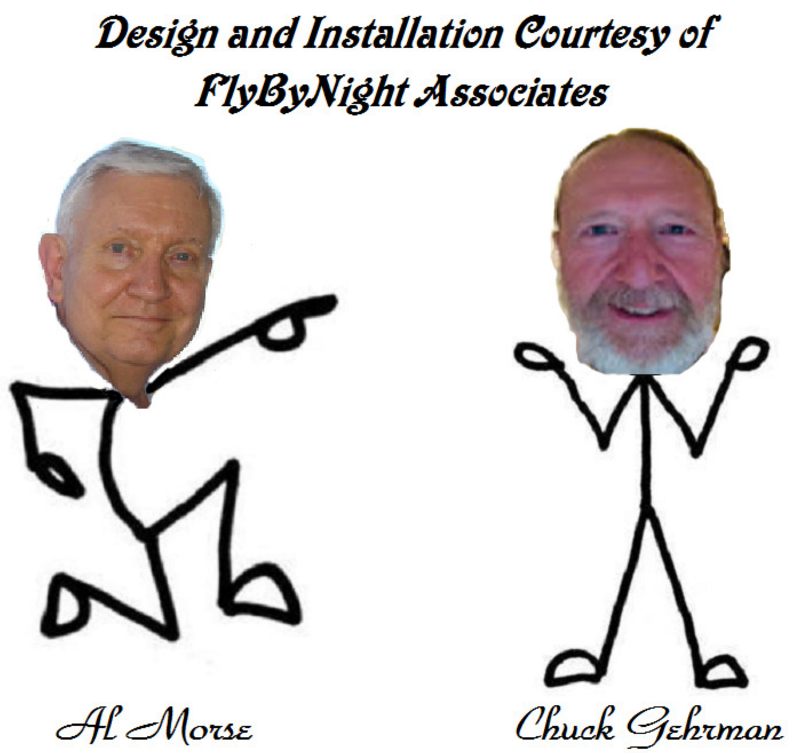
Frequency compensation was derived by broadcasting and recording white noise from each speaker group. An FFT was used to calculate the overall effects of the speakers, their placement in the chambers, and the actual room reflection characteristics. A reverse frequency offset was then applied to all the audio samples using batch processing in an audio editor. The results are very good, and the perceived sound, through all the reflections, speakers, and chamber characteristics, is very realistic.

Because frequency compensation was applied directly to the organ samples, they are highly customized to the CUMC Möller installation and no longer resemble the original samples.

5.0 Credits

Many thanks to Jiri Zurek for kind permission to use dry samples from Sonus Paradisi as the basis of many Möller Extension stops. Thanks to Clinton Scott for his fearless skill at installation of speaker cabinets in elevated, cramped organ chambers.

Otherwise:



Appendix – Hints and Quirks

A1-Setting up the MIDI controls

All the combination pistons, mechanical drawknobs, etc. are fed into the Hauptwerk software through MIDI commands. These commands are "learned" by Hauptwerk and retained, hopefully forever. If the commands are lost for some reason then the learning process must be repeated.

The learning process is covered in the Hauptwerk manual. Each control is placed into a learning mode and the organ hardware control is operated until Hauptwerk says "done". This process *requires* that one, and only one, MIDI command be generated when the Möller control is operated. There are several tricks which must be used to assure this condition. When a piston is learned, you must assure that no digital drawknob is also operated by it. For example, if General 1 is pressed, it must not also engage one of the drawknobs of the digital extension. These are the new stops which have mechanical drawknobs, as well as the Tremulant drawknobs. It might be useful to have the registration selector switch to a memory bank which does not use the digital extensions.

There was a shortage of inputs on the keyboard encoders and all pistons could not be enabled for the digital extensions. The Swell registration pistons are only activated for pistons 1 through 4. Swell pistons 5 and 6 therefore only apply to the original Möller pipe stops.

The keyboards are learned by tapping the bottom most and top most keys. There is one quirk to this installation. The top key of the Choir keyboard needs more filtering on the filter-inverter board. Until a larger capacitor is added, the Choir manual should be learned only from the bottom C to the top B. The digital ranks of the Choir will therefore only play on the bottom 60 keys.

It is best to program the rotary memory bank switch manually as described in Appendix-A3. If that does not work, then special care is required to "learn" the rotary selector switch. The problem is that when the switch is moved from position 2 to 3, for example, it not only engages position 3, but it disengages position 2. This violates the Hauptwerk requirement of a single MIDI command. Therefore, in order to "learn" the rotary switch commands some disassembly of the digital organ circuitry is required.

The top cover of the Möller console needs to be removed to expose the control circuitry.

The rotary switch is shown in figure A-1. The front wafers are used by the Peterson memory select system. The two rear wafers are used by the digital extension circuitry. The rearmost wafer has eight pins which correspond to the eight switch positions. These pins are connected to button inputs on a keyboard encoder. The switch will ground which ever pin corresponds to the switch position.

The black wire on the second most rear wafer is the ground (switch pointer) and must be unsoldered such that none of the eight pins are grounded. ***Be sure to turn the organ off before soldering or de-soldering the black wire!***
The procedure is to place Hauptwerk in the learn mode for position one (Registration: comb. Set: favorite 01) and tap the black ground wire to the pin from which it was soldered. Hauptwerk should say “done”. Then place the rotary switch to position 2, set Hauptwerk to learn favorite set 02, and tap the black ground wire to the wafer pin from which it was unsoldered. Repeat for all eight positions.

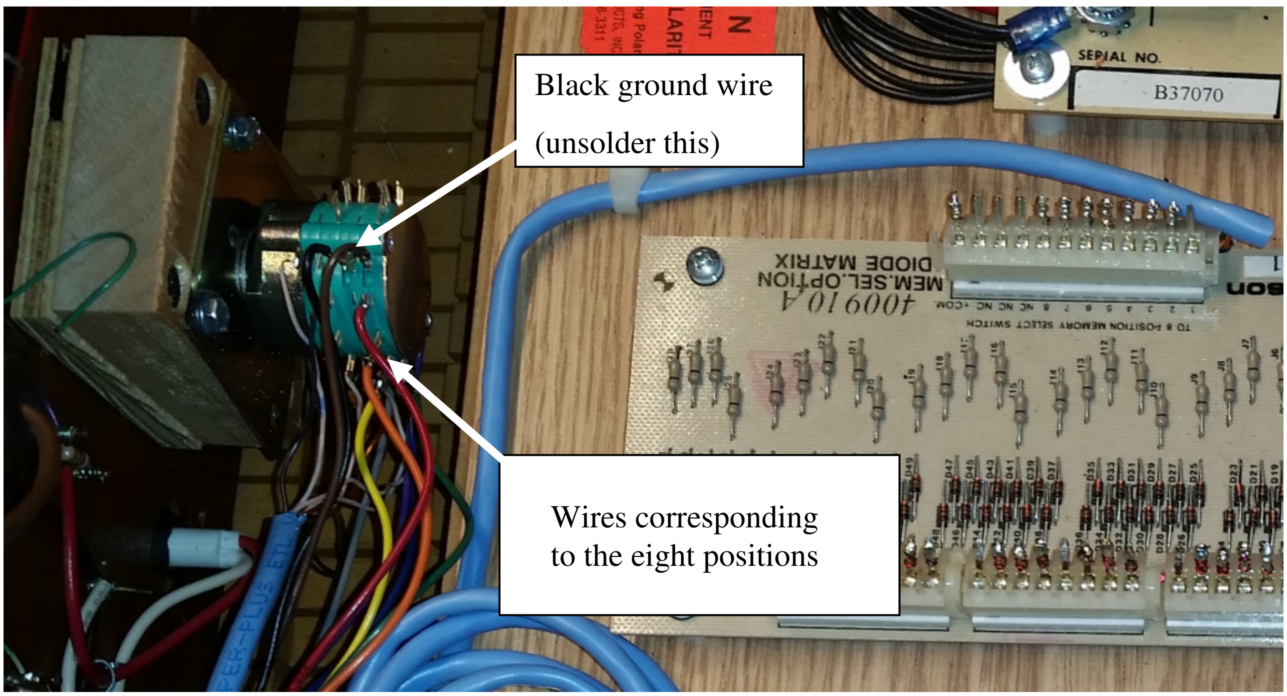


Figure A1 - Rotary selector switch detail

When all eight positions have been “learned” the black wire can be re-soldered. ***Be sure to turn the organ off before soldering or de-soldering the black wire!***

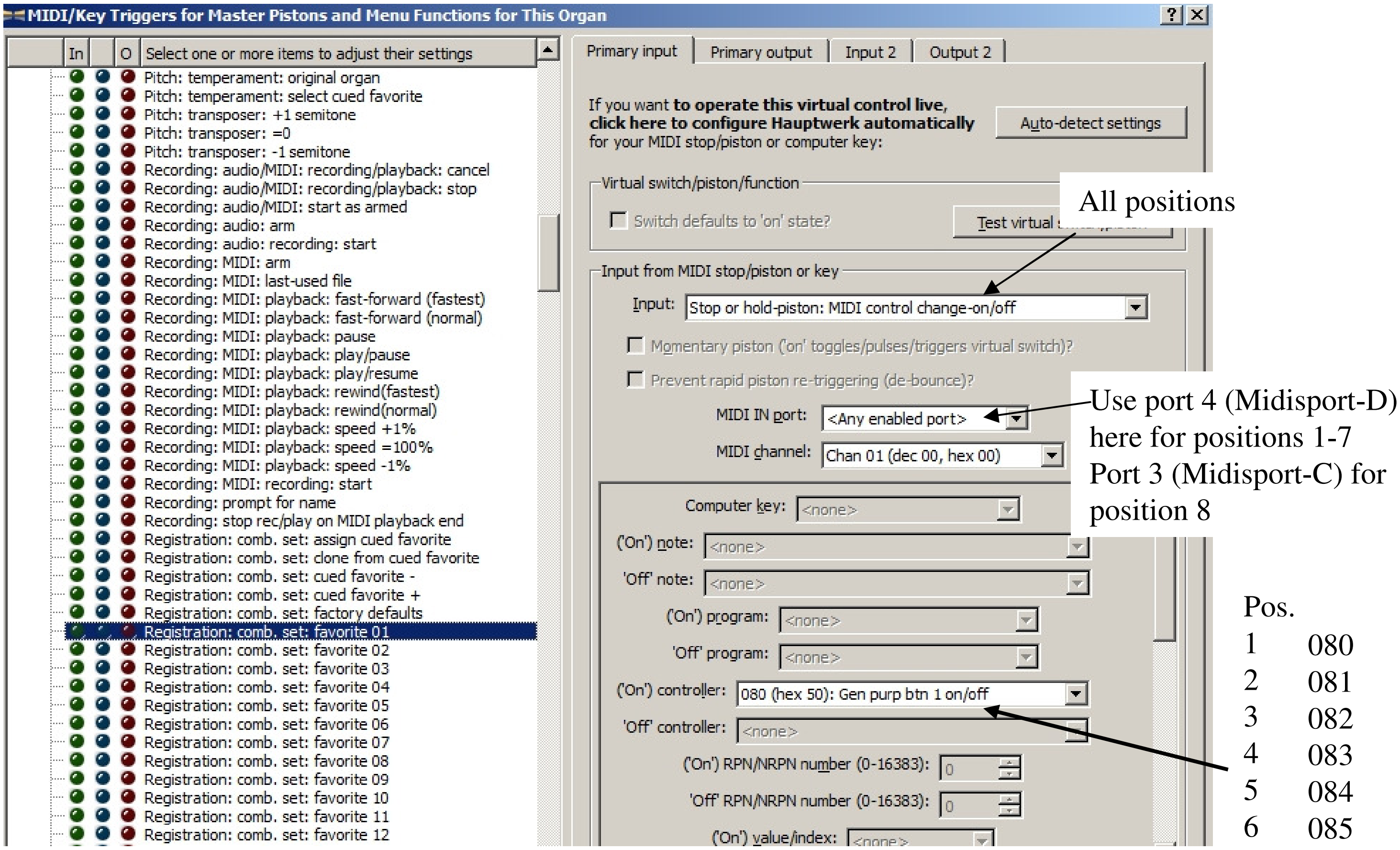
A2-Setting up the temperature control

The automatic temperature tuning is controlled by an analog input to the pedal keyboard encoder. To learn this input, use the Organ Settings Menu –" Expression/crescendo pedals…" input.

For the learn process, move the three position slide switch on the Autotune assembly from one extreme to the other. This can be done several times. After the "done" appears, get out of the learn mode and place the switch in the middle position. Hauptwerk should show that the MIDI range of the control goes from about 38 to 102.

A quirk in Hauptwerk is that when the organ is first loaded, all expression pedal inputs (such as our autotune input) indicate the default value until a change occurs. For the autotune circuit, this means that it will indicate 72° until the temperature sensor changes by at least 0.64°F. The organ may be temporarily out of tune. This quirk is normally self-correcting since the church temperature is not absolutely constant. It is only seen when the MIDI is first learned, or the organ is loaded from scratch, which should happen infrequently. (The proper temperature indication can be sped up by temporarily warming the temperature sensor with the fingers.)

A3 - To set up the rotary switch manually:



Pos.	
1	080
2	081
3	082
4	083
5	084
6	085
7	086
8	080

This is what the SysFunctionSwitchInputOutput tag in the organ config file should look like (compressed)


```
<o><a>323</a></o>
<o><a>1328</a><c>24</c><g>4</g><h>1</h><m>81</m></o>
<o><a>1329</a><c>24</c><g>4</g><h>1</h><m>82</m></o>
<o><a>1330</a><c>24</c><g>4</g><h>1</h><m>83</m></o>
<o><a>1331</a><c>24</c><g>4</g><h>1</h><m>84</m></o>
<o><a>1332</a><c>24</c><g>4</g><h>1</h><m>85</m></o>
<o><a>1333</a><c>24</c><g>4</g><h>1</h><m>86</m></o>
<o><a>1334</a><c>24</c><g>4</g><h>1</h><m>87</m></o>
<o><a>1335</a><c>24</c><g>3</g><h>1</h><m>81</m></o>
<o><a>1336</a></o>
```


This is what the SysFunctionSwitchInputOutput tag should look like (favorite 01, uncompressed)

Tree Selection Browser

SysFunctionSwitchInputOutput

(This tag has no attibutes.)

76 Subtags:

Tag name/Text	Text
ID	1328
DefaultToEngaged	N
Input1_InputTypeID	24
Input1_EngagingEventTogglesOrPulses	N
Input1_PreventRapidReEngaging	N
Input1_EnabledMIDIPortID	4
Input1_MIDIChannelCode	1
Input1_EngagingMIDINoteCode	
Input1_DisengagingMIDINoteCode	
Input1_EngagingMIDIProgChangeCode	
Input1_DisengagingMIDIProgChangeCode	
Input1_EngagingMIDIControllerCode	81
Input1_DisengagingMIDIControllerCode	
Input1_EngagingMIDIRPNorNRPNNumber	0
Input1_DisengagingMIDIRPNorNRPNNumber	0
Input1_EngagingEventValueCode	
Input1_DisengagingEventValueCode	
Input1_ComputerKeyCode	
Input1_RodgersBitfieldStopCode	
Input1_RodgersBitfield2ndStopCode	
Input1_ContentWyvernCCBitfieldStopCodeWithinGroup	0
Input1_ContentWyvernKeyAftertouchBitfieldStopCodeWithinGroup	0
Input1_AhlbornStopGroupCode	

This number is offset by 1 from the setup screen! It must be a bug in Hauptwerk.

mpc64up
Universal UDB–MIDI keyboard encoder

1. What is this?

mpc64up is the user-programmable successor of one of our best-selling units: **mpc64xr** and covers its entire functionality, adding better performance, wider capabilities within smaller size and lower cost.

mpc64up is capable to scan and encode to MIDI up to 64 key contacts or general contacts configured for passive or active keying. It has 7 additional switch/button inputs as well as 8 analog inputs where potentiometers can be wired directly for allowing Continuous control by faders, pedals etc.

Hence, **mpc64up** has the capability of encoding up to 64 + 7=71 contacts and 8 potentiometers. This is enough for most typical keyboard or control applications involving standard 5-octave keyboard and expression controls (faders, pedals).

The most important capability of this unit though is its ***user-programmability***! Each of contacts can be programmed to send any combination of up to 32 MIDI bytes upon engaging and up to 32 MIDI bytes upon releasing. Each of potentiometers can be programmed to send up to 32 MIDI bytes upon position change where one or more of these bytes can carry the data about potentiometer's position (number between 0 and 127).

The support of user-programmability allows using **mpc64up** for **controlling virtually any MIDI-controllable software.**

Last but not least: **mpc64up** has MIDI Merge input capable of full merging, with 256-byte buffer, allowing other software applications to be cascaded to **mpc64up** without additional virtual merging software. The MIDI input is also used for programming MIDI events in **mpc64up.**

2. How it works?

In **mpc64up** internal program memory there is a table of MIDI bytes. For each of key inputs there are two entries/strings containing 32 MIDI-bytes: one string for On event and the other string for Off event. For each of potentiometers there is an 32-byte entry/string containing 32 MIDI bytes that are sent upon each potentiometer position change.

Hence, there is a table of totally 64 + 7 + 8 = 79 MIDI strings, each 32 bytes long.

Each of these strings can be reprogrammed by user for any MIDI content. The string can contain MIDI status bytes, MIDI Data bytes, System and Realtime MIDI messages, etc. It is entirely up to the user what MIDI string will be sent by triggering/changing each of **mpc64up** inputs.

The programming is done by uploading the specific MIDI string to specific table entry via **mpc64up** MIDI input. An special Windows-based application was designed for this purpose, called **mpc64upprg.exe.** It is available for free download on our site (follow links on **mpc64up** product page).

The **mpc64up** can be programmed using other Windows or non-Windows applications, providing that they can send user-defined System Exclusive messages and user-defined MIDI strings. The programming sequence and messages protocol and format are described in ***Appendix A*** of this document.

3. MIDI implementation

Appendix B shows the *factory-programmed* MIDI implementation of **mpc64up.**

This MIDI implementation can be used as it is and can be changed by user when/if necessary. In it each of scan points triggers NoteOn/NoteOff messages on separate channel per scan matrix. MIDI Channel 1 is covered. The note range is MIDI notes 36 - 99. The additional switch/button inputs trigger Control Change messages (CC80-CC86) on MIDI channel 1. The continuous/potentiometer inputs trigger continuous controller CC7 (Volume) on MIDI channels 1-8.

4. Wiring diagrams

Normally, the **mpc64up** is used in systems with passive (GND) keying. It can be confuigured for active positive or active negative keying viltage upon request. Unless other requested the unit is supposed to work with passive keying. *An unit configured for passive keying MUST NOT Be used insystems with active keying. Applying keying voltage in such unit would damage it!* the keying voltage can be configured for Key inouus only!. The additional switch/button inputs cannot be configured for anything than GMD keying. The wiring diagram showing all cases of wiring can be foiund in ***Appendix C.*** The key/switch/button and potentiometer control inputs should be connected to keys/switches/buttons and potentiometers as shown on schematic. Each non-wired key/button/switch input will be read as being open contact (break). Each potentiometer input left unconnected will be read as potentiometer left in Max position. A key/button/switch input connected to GND will be read as being closed contact (make). In case potentiometer input is grounded it will be read as potentiometers left in Min position.

The proper Control Change (or other programmed by user) messages for each button/switch and potentiometer will be transmitted once upon on initializing (starting up).

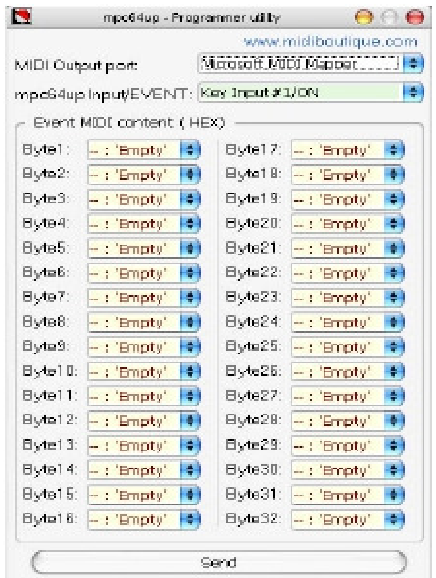
5. Technical specification.

Parameter	Value	Unit	Comment
Power supply voltage	9-12	V	Any adapter/transformer capable to source 50mA or more at 9-12V AC/DC can be used
Power supply current	50	mA	
Number of contact inputs (scan points)	64 + 7 = 71	-	Normally open or normally closed type
Scan rate for contacts	300	S^{-1}	Each key contact is being scanned 300 times per second
Number of analog/potentiometer inputs	8		10 - 100 kOhm linear potentiometers (preferably 10kOhm)
Scan rate or potentiometers	50	S^{-1}	Each potentiometer is being scanned 50 times per second
MIDI messages	User-defined	-	Up to 32-byte MIDI user-defined string per event
MIDI channels	User-defined	-	Defined per MIDI event (single MIDI string can contain MIDI messages going on various MIDI channels
MIDI Merge	Yes	-	256 byte buffer
Size	13.4x8.4x2.5	cm	Approx. 5.3"x3.3"x1"
Weight	70	g	Approx. 2.5 oz

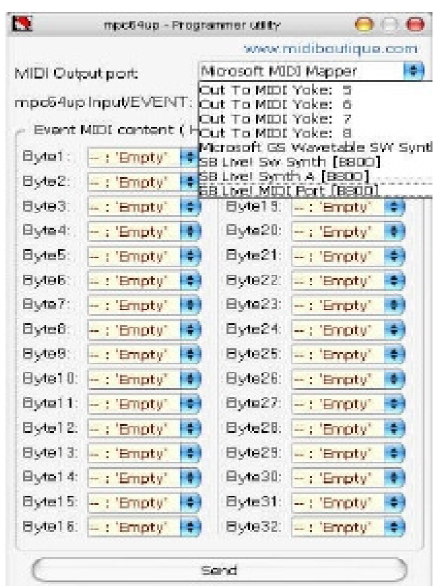
Appendix A. mpc64up – Programming

Step-by-step programming sequence

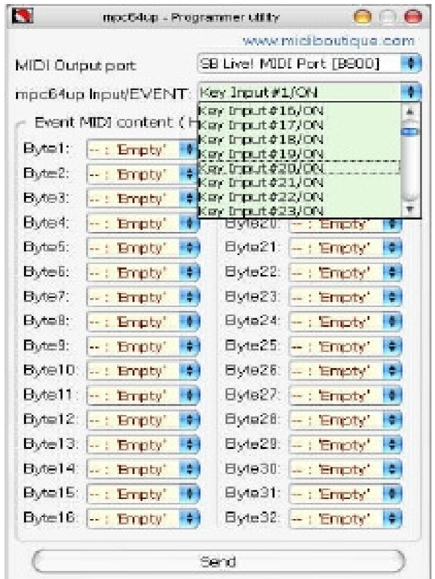
1. Download the ***mpc64upprg.zip*** file from our site (www.midiboutique.com). The actual link can be found on **mpc64up** product page.
2. Copy and unzip the downloaded file to dedicated folder.
3. Connect computer's MIDI output to **mpc64up** MIDI input using standard MIDI cable.
4. Power up the **mpc64up**.
5. Run the unzipped ***mpc64upprg.exe*** utility.



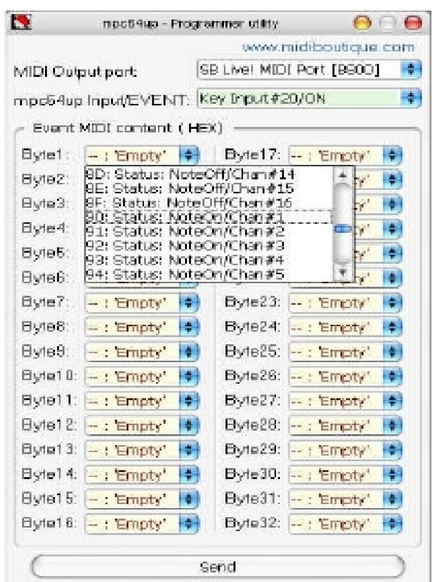
6. Select the proper MIDI output port from *'MIDI Output'* drop-down list.
NOTES:
*Some systems may have more than one MIDI output, there could be hardware and virtual ports as well. Make sure you have selected the hardware port that is connected to **mpc64up** in previous steps.*



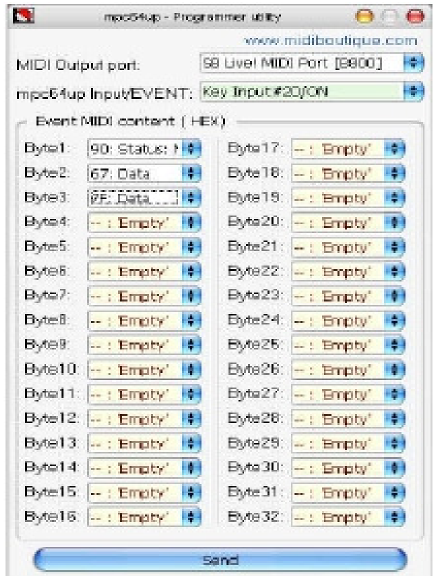
7. Select the event you want to program. All user programmable MIDI events are listed in '**mpc64up** Input/EVENT' drop-down list. There are three groups of events: ON-event for contact inputs, OFF-event for contact inputs and CHANGE events for potentiometer inputs.



8. Choose desired event content byte-by-byte by selecting byte values.
NOTES:
Bytes that have assigned 'Empty' will be reset to value of FF.
For any event, each byte that has been assigned HEX value of FF will be ignored and won't be producing MIDI output traffic.
For potentiometer events each byte that has been assigned HEX value of F6 will be replaced by potentiometer data in range 0-127 as read from potentiometer.
*Hence, **System Reset** MIDI message (HEX FF) cannot be programmed in any event and **Tune Request** MIDI message (HEX F6) cannot be programmed in potentiometer events.*



9. After the MIDI string has been configured, press once the 'Send' button at the bottom.



10. Repeat steps 6 .. 9 as many times as necessary for programming desired events.
11. The unit is programmed and can be used.

NOTES:
The programmed MIDI strings will be permanently kept in non-volatile (power-independent) memory and won't change until next programming.
*More than one **mpc64up** unit can be chained together and will be programmed simultaneousness as each **mpc64up** would retransmit whatever MIDI traffic it gets.*

Programming protocol

The programming protocol includes three parts:

Header message.
This is *optional* 10-byte System exclusive message that only causes reset of program input queue (the 32-byte input buffer that receives the MIDI string to be programmed). During reset, all the 32 bytes of Program buffer are set to HEX value of FF. If the string to be programmed is 32 bytes long, the header message can be omitted as the input queue will be updated entirely . The format of this message is:

Header message - 10 bytes (all shown in HEX format)

```
F0 - SysEx start
00 - first ID
21 - sec. ID  (MGB)
7F - thd. ID   (MGB)
0D - Device ID (mpc64up)
00 - Device sub-ID (message:   Reset buffer)
xx - future use byte (set to   00)
xx – future use byte (set to   00)
xx – future use byte (set to   00)
F7 - SysEx end
```

The MIDI string itself.
It can be between 0 and 32 bytes long. In case of 0 bytes, the MIDI string for the programmed event is considered empty and this event won't be producing any MIDI output. This has the same effect as programming HEX FF value to all 32 bytes of this

string. This feature is suitable for programming switches that would transmit Program change messages upon contact make and nothing upon contact break.
If an MIDI string is longer than 32 bytes, only the last 32 bytes will take place. Any MIDI bytes of any order can be transmitted. The only two values that have special meaning are HEX FF (MIDI Reset) and HEX F6 (Tune Request). These are used for inserting special parameters in MIDI string. Refer to notes after the Step 8 of step-by-step programming sequence described above about these special considerations.

Footer message.
This is *obligatory* 10-byte System exclusive message that passes to **mpc64up** the table entry number to be programmed, and invokes the memory-write routines to copy the input buffer content to proper table entry.

Footer message - 10 bytes (all shown in HEX format)

```
F0 - SysEx start
00 - first ID
21 - sec. ID  (MGB)
7F - thd. ID
0D - Device ID (mpc64up)
00 - Device sub-ID (message ID = 01: Store buffer)
ll - Entry number, LSB (7-bit value 00..7F)
mm - Entry number, MSB (7-bit value 00..07)
xx - Future use byte (set to 00)
F7 - SysEx end
```

Appendix B. mpc64up – factory MIDI implementation

Keys

Key#	Control type	MIDI message on make	MIDI message on break	MIDI channel	Comment
1	momentary contact/switch	NoteOn #36	NoteOff #36	1	
2	momentary contact/switch	NoteOn #37	NoteOff #37	1	
3	momentary contact/switch	NoteOn #38	NoteOff #38	1	
4	momentary contact/switch	NoteOn #39	NoteOff #39	1	
5	momentary contact/switch	NoteOn #40	NoteOff #40	1	
6	momentary contact/switch	NoteOn #41	NoteOff #41	1	
7	momentary contact/switch	NoteOn #42	NoteOff #42	1	
8	momentary contact/switch	NoteOn #43	NoteOff #43	1	
9	momentary contact/switch	NoteOn #44	NoteOff #44	1	
10	momentary contact/switch	NoteOn #45	NoteOff #45	1	
11	momentary contact/switch	NoteOn #46	NoteOff #46	1	
12	momentary contact/switch	NoteOn #47	NoteOff #47	1	
13	momentary contact/switch	NoteOn #48	NoteOff #48	1	
14	momentary contact/switch	NoteOn #49	NoteOff #49	1	
15	momentary contact/switch	NoteOn #50	NoteOff #50	1	
16	momentary contact/switch	NoteOn #51	NoteOff #51	1	
17	momentary contact/switch	NoteOn #52	NoteOff #52	1	
18	momentary contact/switch	NoteOn #53	NoteOff #53	1	
19	momentary contact/switch	NoteOn #54	NoteOff #54	1	
20	momentary contact/switch	NoteOn #55	NoteOff #55	1	
21	momentary contact/switch	NoteOn #56	NoteOff #56	1	
22	momentary contact/switch	NoteOn #57	NoteOff #57	1	
23	momentary contact/switch	NoteOn #58	NoteOff #58	1	
24	momentary contact/switch	NoteOn #59	NoteOff #59	1	
25	momentary contact/switch	NoteOn #60	NoteOff #60	1	
26	momentary contact/switch	NoteOn #61	NoteOff #61	1	
27	momentary contact/switch	NoteOn #62	NoteOff #62	1	
28	momentary contact/switch	NoteOn #63	NoteOff #63	1	
29	momentary contact/switch	NoteOn #64	NoteOff #64	1	
30	momentary contact/switch	NoteOn #65	NoteOff #65	1	
31	momentary contact/switch	NoteOn #66	NoteOff #66	1	
32	momentary contact/switch	NoteOn #67	NoteOff #67	1	
33	momentary contact/switch	NoteOn #68	NoteOff #68	1	
34	momentary contact/switch	NoteOn #69	NoteOff #69	1	
35	momentary contact/switch	NoteOn #70	NoteOff #70	1	
36	momentary contact/switch	NoteOn #71	NoteOff #71	1	
37	momentary contact/switch	NoteOn #72	NoteOff #72	1	
38	momentary contact/switch	NoteOn #73	NoteOff #73	1	
39	momentary contact/switch	NoteOn #74	NoteOff #74	1	
40	momentary contact/switch	NoteOn #75	NoteOff #75	1	
41	momentary contact/switch	NoteOn #76	NoteOff #76	1	
42	momentary contact/switch	NoteOn #77	NoteOff #77	1	
43	momentary contact/switch	NoteOn #78	NoteOff #78	1	
44	momentary contact/switch	NoteOn #79	NoteOff #79	1	
45	momentary contact/switch	NoteOn #80	NoteOff #80	1	
46	momentary contact/switch	NoteOn #81	NoteOff #81	1	
47	momentary contact/switch	NoteOn #82	NoteOff #82	1	
48	momentary contact/switch	NoteOn #83	NoteOff #83	1	
49	momentary contact/switch	NoteOn #84	NoteOff #84	1	
50	momentary contact/switch	NoteOn #85	NoteOff #85	1	
51	momentary contact/switch	NoteOn #86	NoteOff #86	1	

52	momentary contact/switch	NoteOn #87	NoteOff #87	1	
53	momentary contact/switch	NoteOn #88	NoteOff #88	1	
54	momentary contact/switch	NoteOn #89	NoteOff #89	1	
55	momentary contact/switch	NoteOn #90	NoteOff #90	1	
56	momentary contact/switch	NoteOn #91	NoteOff #91	1	
57	momentary contact/switch	NoteOn #92	NoteOff #92	1	
58	momentary contact/switch	NoteOn #93	NoteOff #93	1	
59	momentary contact/switch	NoteOn #94	NoteOff #94	1	
60	momentary contact/switch	NoteOn #95	NoteOff #95	1	
61	momentary contact/switch	NoteOn #96	NoteOff #96	1	
62	momentary contact/switch	NoteOn #97	NoteOff #97	1	
63	momentary contact/switch	NoteOn #98	NoteOff #98	1	
64	momentary contact/switch	NoteOn #99	NoteOff #99	1	

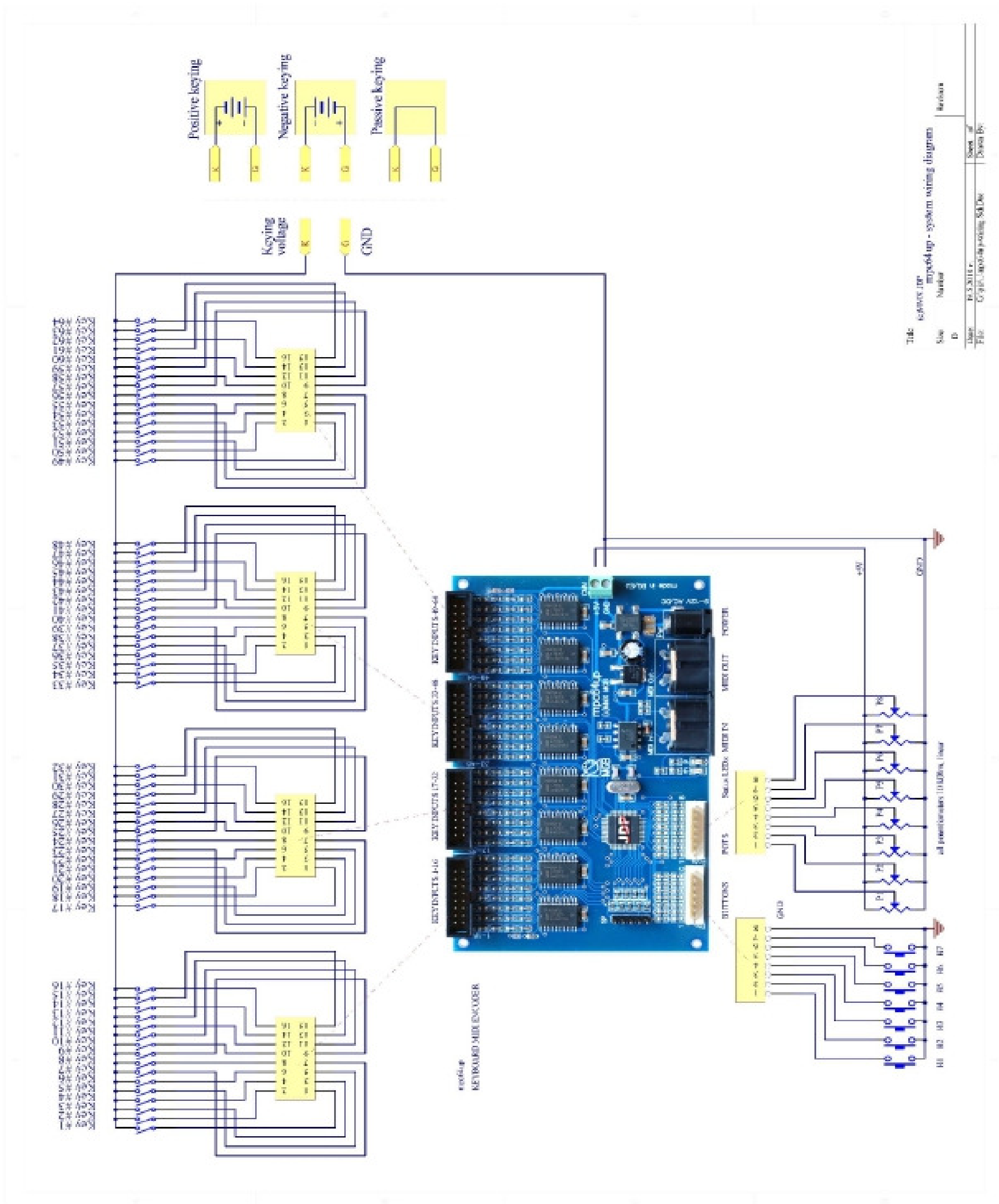
Buttons

Button#	Control type	MIDI message on make	MIDI message on break	MIDI channel	Comment
1	momentary contact/switch	CC#80 On	CC#80 Off	1	
2	momentary contact/switch	CC#81 On	CC#81 Off	1	
3	momentary contact/switch	CC#82 On	CC#82 Off	1	
4	momentary contact/switch	CC#83 On	CC#83 Off	1	
5	momentary contact/switch	CC#84 On	CC#84 Off	1	
6	momentary contact/switch	CC#85 On	CC#85 Off	1	
7	momentary contact/switch	CC#86 On	CC#86 Off	2	

Potentiometers

Pot#	Control type	MIDI message on change	MIDI channel	Comment
1	potentiometer or control voltage 0 .. +5V	CC7 (Volume)	1	
2	potentiometer or control voltage 0 .. +5V	CC7 (Volume)	2	
3	potentiometer or control voltage 0 .. +5V	CC7 (Volume)	3	
4	potentiometer or control voltage 0 .. +5V	CC7 (Volume)	4	
5	potentiometer or control voltage 0 .. +5V	CC7 (Volume)	5	
6	potentiometer or control voltage 0 .. +5V	CC7 (Volume)	6	
7	potentiometer or control voltage 0 .. +5V	CC7 (Volume)	7	
8	potentiometer or control voltage 0 .. +5V	CC7 (Volume)	8	

Appendix C. Wiring diagram



NOTE:
More detailed copy of this diagram is available on our site in **.pdf** format!

info@midiboutique.com
www.midiboutique.com