

**USER MANUAL**  
for  
PEDSCAN rev. C – MIDI controller  
firmware version 1.7

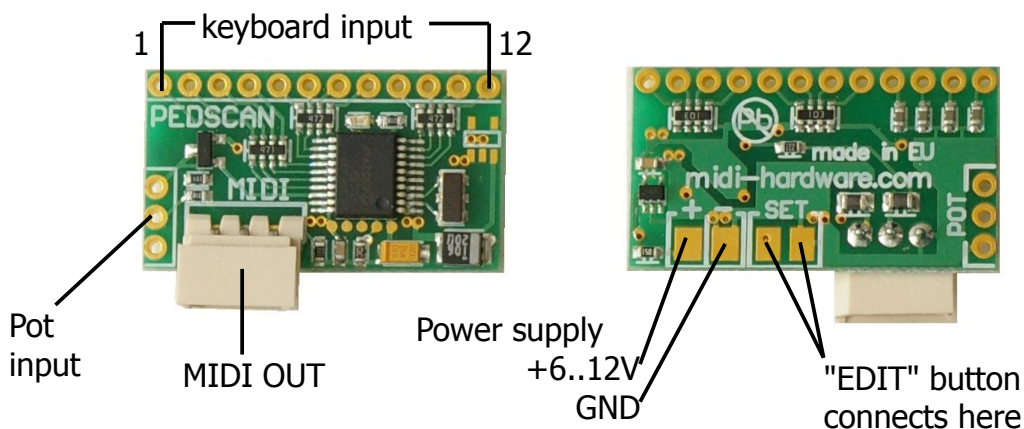
# 1 Overview

This little board is standalone controller for 32 keys and 1 potentiometer or external voltage input. Contact closing or potentiometer movement is converted into MIDI. This board is most usable when no more than 32 keys are needed, like organ pedals, but it can also work as MIDI parameter pedal, like Pitch Bend etc. The board comes in 2 versions. One for use with pedal boards, or button-boxes. It works with 32 keys and 1 analog (pot) input. The other version is just single potentiometer MIDI converter. It has only one potentiometer input and no keyboard can be used. It is useful for MIDI pedals, like volume etc. To easily setup all the features, it comes with numeric keypad soldered directly to the board.

# 2 Features

- scans 32 contacts organized in diode matrix
- 1 analog input for potentiometer, or external voltage
- user defined MIDI channel separately for each keyboard split and pot
- user defined MIDI event separately for each split of the keyboard and the pot
- available knob and keyboard events: Notes, Control Change, Pitch Bend, Channel After Touch, Program Change, special modes for Ahlborn, MidiTzer, NI B4
- user defined transposition independent for each keyboard split
- all settings stored in nonvolatile memory
- requires 5V DC power supply, about 1mA current consumption depending the value of pot used.
- optional auto-power-off feature for battery powered system.

# 3 Layout



## 4 Power supply

Recommended power supply range is between 5 and 12V DC. It is possible to run this board from lower voltage, but its operation is not guaranteed then. Current consumption is about 1mA so it can be used with typical 9V battery for many hours. When using potentiometer, additional current determined by potentiometer value will add to overall current drain. If e.g. 10k pot is used, current consumption will rise by 0.5mA. Power input is located at the bottom, there are 2 rectangular solder pads as shown in Layout section before. Positive terminal (+) is closer to the left edge, while negative terminal (-) is closer to 2 other similar looking rectangular pads. Use thin stranded wires to connect the power. This is delicate device and power pads cannot be soldered many times. Try to do it right for the first time. **Connecting power in reverse will permanently damage the board, and will void the warranty.**

### 4.1 Auto power off (optional)

If requested, this board has the auto-power-off feature enabled. It means it can be connected to a battery pack without power switch, and in periods of no use it will preserve the battery energy. After last key was released, it waits about 20 minutes and then enters the low power mode. Then it draws minimal current (about 10uA), small enough not to significantly affect battery life. It goes back to normal full speed mode after any note is played, or potentiometer moved. After a short delay of 16ms (0.016s, not noticeable by most people) it plays MIDI note that was pressed on keyboard. This can be described as increased keyboard latency for the first note played after quitting the low-power mode. During normal operation the keyboard latency is less than 3ms. It is also possible to wake it up by turning the potentiometer if it is installed. The board wakes after maximum delay of 1s after the pot was moved.

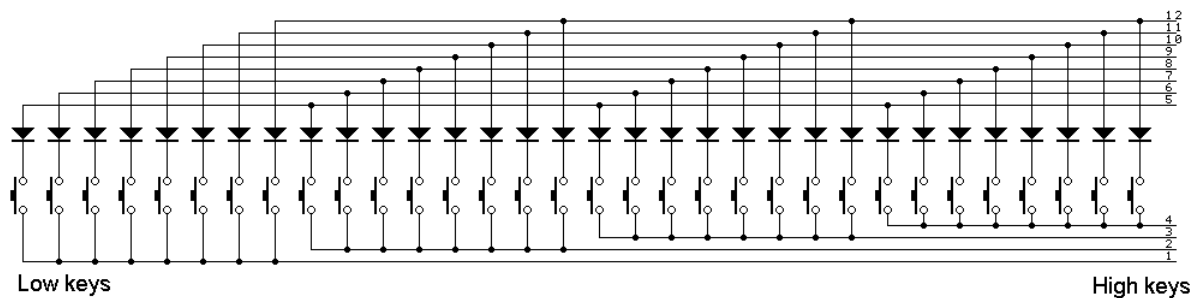
After another 4 hours of idle, the boards enters power-off mode, further reducing current down to about 5uA. This is comparable to selfdischarge rate of typical alkaline AAA batteries, so if PEDSCAN is not used for long periods of time, you don't need to bother taking the power off. In this state it does not react to potentiometer movement at all, and to wake it up you must play any note and keep the key pressed for at least half a second. A MIDI note will play, and PEDSCAN is fully up again.

## 5 Connecting keyboard

Keyboard switches are connected via diode matrix organized in 8 rows and 4 columns. The board itself is only a matrix driver and can be used with bare contact only if it's to be used as monophonic keyboard, or some kind of selector (like Program Change buttons). If the board will be used polyphonically, with more than 1 key pressed at any time, proper diode matrix is required, otherwise you'll get more notes on MIDI than you'd ever want.

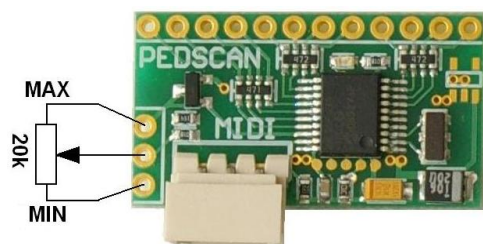
### 5.1 What is a "diode matrix"?

8x4 scanner driver can be used if the keyboard has "scanning diode matrix", that's special kind of very simple circuit, made of diodes forming electric XY array of 8 rows and 4 columns. It's easy to make the matrix on your own, if you know how to solder, and follow electrical schematic. It works best if it's built directly on contacts, then the overall length of wire used is the shortest. That also improves keyboard performance. Below is the schematic of compatible diode matrix



## 6 Connecting potentiometer

Pot input can be used as continuous controller for things like volume, modulation etc. Usually this input would be connected to a potentiometer, but it's possible to use it as analog input with range of 0..+5V. Applying voltage of 0V causes PEDSCAN to generate CC with lowest value, while +5V generates highest possible value of assigned MIDI parameter. This input is fully user configurable in terms of controller type and MIDI channel. Potentiometer must be linear taper (not audio) in range 5-50k, preferably 20k. The picture above shows how to connect the potentiometer, and which side corresponds to maximum MIDI value, and which side to minimum.



Other values of potentiometer are of course possible, but using pot lower than 10k will cause more current consumption, important in battery powered controller. Values higher than 50k will exhibit nonlinear function. i.e. when the pot is in the center position, resulting MIDI parameter value will not be half of maximum, but

slightly more. The bigger pot value, the more non-linear position/value behavior is apparent.

## 7 MIDI settings and special functions

All settings are performed from connected keyboard. There is one special input on the board (2 pads marked on first page as EDIT button input) for entering the EDIT mode, and all parameters can be changed then. In special 1-pot-only version with no keyboard, but with numeric keypad soldered, the EDIT button is the "#" button on the keypad. The keypad version also has one button - "\*" which is used to select one of the 10 favorite patches (Program change).

To change any settings, you have to enter new value of given parameter. To do so, use lowest 10 keys of the keyboard as numeric entry. Lowest key is digit "0", while 10<sup>th</sup> key is digit "9". This is obvious when the keypad is used. As a general rule, any change on a controller requires selecting this controller first before making change. For example, if you want to change MIDI channel of potentiometer, move it a bit, and go into MIDI channel settings mode. Or to change the split point – first play a note on the keyboard, and enter split-point change mode.

Numeric entries are provided with the assumption that numeric keypad is used, but the same can be achieved with lowest 10 keys of any keyboard connected to PEDSCAN. It helps to add a sticker over lowest 10 keys with numbers from 0 to 9 if only musical keyboard is used.

In this chapter, describing how to set all parameters, whenever „#" sign is mentioned, it means the "EDIT" key, which is located at „#" key on numeric keypad, or 2 pads for EDIT button on the bottom side of the board. Likewise, "Patch Recall" button is called „\*" and it is „\*" key on keypad, but it's not available in 32-keys version.

### 7.1 Transposition of keyboards

Transposition of keyboards connected to PEDSCAN is unlimited, that means any key can generate any MIDI note from range of over 12 octaves. There are two ways of using it. Typical one is by selecting "new middle C" position. First you have to select the keyboard you want to edit by playing any note on it. Enter „#" then „1" on the keypad. Now, whatever key you press, it will be the new position of the middle C MIDI note afterwards. You can select new position of middle C note anywhere between 3<sup>rd</sup> and top key of the keyboard.

Another option is to use lowest 2 keys of the keyboard, or numbers "0" and "1" of the keypad. It doesn't matter if keyboard starts with key C or F or whatever, those are always two lowest keys. The 1<sup>st</sup> one shifts the keyboard one semitone down with each sequence (**#10**), the 2<sup>nd</sup> shifts the keyboard one semitone up (**#11**). This is useful when you want to shift the keyboard in range not available by the first method. Both methods require first selecting the keyboard to be changed by playing a note, then entering "# 1" on the keypad, and selecting transposition.

## 7.2 MIDI event assignment for keyboards and analog inputs

MIDI event assigned to given potentiometer or keyboard split can be easily changed. To perform this, turn a bit the knob, or play a key on the split you want to assign, and then select the controller type by entering keys **#** then **2** and then appropriate number from table in Appendix A. You need to enter 2 or 3 digits for each input controller depending on entered number. Possible MIDI event codes are from number 000 to 149. **Standard setting for a keyboard is "# 2 131" - single notes, and for analog input "# 2 007"** - that's MIDI volume. This is factory default. Possible settings described as follows.

### 7.2.1 Control Change - #2 CC

Any MIDI Control Change number in range from "000" till "127". Numbers above 127 are used to generate MIDI events other than *Control Change*, or turn them into other functions, what is described next.

### 7.2.2 Pitch Bend - #2128

The pot will work then as pitch bender. If assigned to a keyboard, each key will set pitch bender in 1/128 steps across the keyboard. Range can be adjusted with transposition settings.

### 7.2.3 Program Change - #2129

Although this is rather unusual usage, this pot will then generate MIDI Program Change messages with its every move. Program Change can be also generated from the keyboard, by using sequence "**# 4** <number>" – this is described later. If assigned to a keyboard, pressing each key will generate MIDI Program Change message with different patch number. Starting number can be adjusted with transposition setting. This is useful for organ emulators, where bank of Program Change buttons can be used to work as pistons (sets of registers)

### 7.2.4 Channel After Touch - #2130

Turning such pot will cause Channel After Touch messages to be sent out. If assigned to a keyboard, each key will set After Touch in 1/128 steps across the keyboard. Range can be adjusted with transposition settings.

### 7.2.5 Standard keyboard action – single notes - #2131

Whenever MIDI event 131 is assigned to a keyboard, it works as typical MIDI keyboard, playing MIDI notes. It is also possible to generate notes played in glissando, when this event is assigned to a pot. Select the pot to be edited, enter "**# 2 131**". This knob becomes then a note generator resembling quantized Theremin. Move the knob and a series of notes will be played. There's only one note played at a time (with velocity set like described later) and it is released just before new note is about to play. Whole knob slow rotation plays 128 notes from entire MIDI range.

### 7.2.6 **Note on - #2132**

This mode is somehow similar to the mode described above, but only "note-on" messages are generated, that means whenever you move this pot, new notes will be played, and they will stay on forever unless proper note-off message will be issued by another means. If assigned to a keyboard, only note-on messages will be sent. It will work like with constantly depressed sustain pedal.

### 7.2.7 **Note off - #2133**

This is like "note-on" mode described above, but instead it sends out only note-off messages. It can be used to mute part of notes already played, or as some kind of panic button – slow full rotation mutes all notes in assigned channel. If assigned to a keyboard, it will send only note-offs, so it may be used to quiet some notes played earlier.

### 7.2.8 **One-touch Patch Recall - #2134**

Keyboard in this mode serves as an array of favorite patches buttons. Each key recalls Program Change (or in another words - selects a patch/preset) that was earlier programmed. There's 32 memory locations, as many as keys. For example you can program key 1 to send Program Change 37, key 2 as PC#76, key 3 as PC#20 etc. Assigning Program Change numbers to a specific key is described later, in paragraph "Favorite Patches".

### 7.2.9 **CC keyboard - #2135**

This feature has no effect on a pot, i.e. the pot will generate no MIDI event if it has this feature assigned. In this mode you can use keyboard as toggle switches selecting min/max values of range of CCs. All keys have increasing MIDI Continuous Controller assigned. Pressed key sends CC with max value (127), while key release generates the same CC but with minimum value (0). CCs are ordered just like there would be MIDI notes, i.e. typically they start from CC#36 at the lowest key, next key is CC#37 etc. Use transposition settings to set different starting CC.

### 7.2.10 **MIDI channel selector for all controls - #2136**

This is usable only with keyboard scanners, because using it with pot needs high precision, as full change takes about 1/8 of a turn. When assigned to a keyboard, first 16 keys work like MIDI channel selector for all controllers. After one of the keys is hit, notes played on another keyboard are played on changed channel. Individual channel settings for every keyboard split and potentiometer described in chapter 7.3 work together with this setting. For example if one split was set to channel 3, and you change the channel using this feature to +4 (by hitting 5th key), resulting channel is 7 (3+4). If all controllers are set to channel 1, then all 16 keys assigned to this feature are direct channel selectors from 1 to 16.

### 7.2.11 **Small Transposer - #2137**

Select the pot to be edited, enter "**# 2 137**". Turning such pot will shift all notes played on all connected keyboard scanners by number of semitones determined by

pot position. In the middle it gives no shift, and full rotation has range from -4 to +4 semitones. It's most useful when pot is replaced by 9-position switch with 8 resistors of equal value connected between switch leads. Assigning this to a keyboard is also possible, but you cannot reach full range of transposition with 5-octave keyboard.

#### **7.2.12 *Big Transposer - #2138***

Turning such pot will shift all notes played on all connected keyboard scanners by number of semitones determined by pot position. In the middle it gives no shift, and full rotation has range from -8 to +8 semitones. Assigning this to a keyboard is also possible, but you cannot reach full range of transposition with 5-octave keyboard.

#### **7.2.13 *Velocity - #2139***

Position of this pot will then determine velocity parameter of all MIDI notes generated by this board, and all connected expanders/scanners. If assigned to a keyboard, each key will set velocity of all notes in 1/128 steps across the keyboard. Range can be adjusted with transposition settings.

#### **7.2.14 *Native Instruments B4 chorus/vibrato - #2140***

Position of this pot will be then reflected in B4 as "chorus/vibrato" switch position. It has only 6 positions, and appropriate command will be sent to B4 every time the potentiometer crosses each position representing another switch position.

#### **7.2.15 *MidiTzer stops control - #2141***

When assigned to a keyboard, each key becomes specific stop controller. When a key is pressed, MIDI controller 81 (51 hex) is sent, and when it is released, MIDI controller 80 (50 hex). Value of the controller is determined by the button pressed. This is default way of controlling stops in MidiTzer organ software.

#### **7.2.16 *Ahlborn Archive module stops control - #2142***

When assigned to a keyboard, each key becomes specific stop controller in Ahlborn Archive organ sound-module. When a key is pressed, MIDI controller 73 (49 hex) is sent, and when it is released, MIDI controller 74 (4A hex). Value of the controller is determined by the button pressed. This is default way of controlling stops in Ahlborn Archive module.

#### **7.2.17 *Ahlborn Organs stops control - #2143***

When assigned to a keyboard, each key becomes specific stop controller in Ahlborn Organs. When a key is pressed, MIDI controller 70 (46 hex) is sent with bit 6 of the value set, and when it is released, the same MIDI controller but with bit 6 of the value cleared. Other bits of the value are determined by the button pressed. In another words, pressing the button sends CC 70 with value range 0-63, and releasing a button - CC 70 with value range 64-127. This is default way of controlling stops in Ahlborn Organs.



### 7.2.18 **One time Velocity setup - #2149**

This is similar to velocity settings described in 7.2.13, but in addition, it saves last used velocity settings in nonvolatile memory. This should be used only during installation, when you want to set default velocity of notes after each power up. For expression and frequent usage, control the velocity by assigning #2139 instead.

## 7.3 **MIDI Channel**

Channel of each potentiometer and keyboard can be set individually. To change MIDI channel of certain potentiometer select the pot by simply turning it a bit. Then select the MIDI channel with the following sequence: "**# 3** <channel number>". The channel number must be in range 1-16. Channels from 2 to 9 require only 1 key stroke, while 1 needs to be entered as 2 digits – namely "01". Channel 10 and above of course need 2 keystrokes too. Then turn another potentiometer and select its channel the same way. To change MIDI channel of the keyboard, first select it by playing a note, and enter mentioned sequence with channel number.

## 7.4 **Program Change**

This device allows you to send Program Change MIDI messages, or in another words – change patches. Three ways are available. Two were mentioned in Knob Assignment chapter, where you could program the potentiometer to act like 128-position patch rotary switch, or use keyboard assigned to Program Change to act like array of single touch patch select buttons. To change the patch on selected keyboard directly to specific number, play a note on this keyboard and enter the sequence: "**# 4** <program number>". The Program Change MIDI message is sent directly after last digit of entered patch number. This may happen after 2<sup>nd</sup> or 3<sup>rd</sup> digit. You only need to enter 2 digit, when the patch number is in range 13-99. Programs lower than 13 require 3 digits, with 0s in front, for example 012, or 003. Obviously, programs with numbers higher than 99 also need 3 digits. The range of Program Change is from 000 to 127.

## 7.5 **Keyboard split**

It is possible to split each keyboard into 2 independent parts. The split point can be anywhere on the keyboard, and both parts can work with independently adjusted MIDI channel, type of event and starting note (transposition) or range of other controllers if something else than notes is assigned to a keyboard. To set up the split point, you have to select the keyboard by playing a note in it, and then enter sequence „**# 5**” followed by stroke of the key that you want to be the last one of the lower part. Since then lower part remains on the same channel that was used for whole keyboard, while upper part takes settings of upper part, which by default is 1 MIDI channel higher. To change MIDI channel, type of event, transposition, or send a Program Change for split part, follow directions described above, regarding

non-split keyboard, but now changes are made to this split part, which was selected by playing a note prior entering the edit mode (pressing „#“).

## 7.6 Favorite patches

Whenever a MIDI program/patch/instrument selected from this controller is often used, or interesting, it is worth to memorize for fast recall in the future. There can be 50 such favorite patches, selected for last played keyboard, split, or turned pot by just 2 key strokes. First you play on a keyboard, and to change a patch for this keyboard, simply press „\*” and then a single key with memorized patch. This key can be any of 50 lowest keys or any keyboard, or digit 0-9 from numeric keypad. The device will send the patch number in channel of last played keyboard, or last turned pot. To memorize any patch for recall in this way, you have to first select this patch by one of the 3 ways – either by entering „# 4 <patch number>”, or by turning „Program Change” knob if one is assigned, or by using one of the keys in keyboard assigned to Program Change event. Then simply press “# 6”, then the key where the patch should be stored and it's done. Next time whenever you press „\*” followed by that digit you just programmed, the MIDI Program Change message will be issued, setting the patch that was programmed into that digit. If you program MIDI event 134 on a keyboard (described in 5.2.8), you don't even have to press “\*” to recall patch. Just hit any key and it's selected.

## 8 Analog input update rate

The analog input of PEDSCAN translates input voltages, or potentiometer position into MIDI. Actual pot position is updated via MIDI every time it changes. This update is however not immediate - this is common to any MIDI knob box. The fastest response for potentiometer movement is about 2.5ms. It means that when you constantly move the pot, PEDSCAN will issue a MIDI update every 2.5ms. This is more than enough for most of uses. In some instruments, either hardware, or virtual, some problem may occur when there is a lot of MIDI traffic. It is also sometimes desirable to limit MIDI traffic e.g. to minimize the size of MIDI file recorded on a sequencer. It is possible to change this setting using command “#98n” in the way described in chapter 7, where “n” determines update rate according to the table below. Default factory setting is 9.5ms.

keypad sequence	#980	#981	#982	#983	<b>#984</b>	#985	#986	#987	#988	#989
minimum response [ms]	2.5	3.5	4.5	6.5	<b>9.5</b>	14	20	30	50	66
update rate [Hz]	400	290	220	150	<b>105</b>	70	50	33	20	15

**Appendix A** – numbers for assigning MIDI event type to potentiometers. For up to date assignment of CC numbers to controllers go to [www.midi.org](http://www.midi.org). Note that features described in *italic* with numbers above 127 don't represent a CC, but other MIDI events or parameters.

digits	Controller name	digits	Controller name
000	Bank Select	68	Legato Footswitch
001	Modulation Wheel or Lever	69	Hold 2
002	Breath Controller	70	Sound Controller 1 (Sound Variation)
004	Foot Controller	71	Sound Controller 2 (Timbre/Harmonic Intens.)
005	Portamento Time	72	Sound Controller 3 (Release Time)
006	Data Entry MSB	73	Sound Controller 4 (Attack Time)
007	Channel Volume	74	Sound Controller 5 (Brightness)
008	Balance	75	Sound Controller 6 (Decay Time)
010	Pan	76	Sound Controller 7 (Vibrato Rate)
011	Expression Controller	77	Sound Controller 8 (Vibrato Depth)
012	Effect Control 1	78	Sound Controller 9 (Vibrato Delay)
013	Effect Control 2	79	Sound Controller 10 (default undefined)
16	General Purpose Controller 1	80	General Purpose Controller 5
17	General Purpose Controller 2	81	General Purpose Controller 6
18	General Purpose Controller 3	82	General Purpose Controller 7
19	General Purpose Controller 4	83	General Purpose Controller 8
32	LSB for Control 0 (Bank Select)	84	Portamento Control
33	LSB for Control 1 (Modulation Wheel or Lever)	91	Effects 1 Depth (Reverb Send Level)
34	LSB for Control 2 (Breath Controller)	92	Effects 2 Depth
36	LSB for Control 4 (Foot Controller)	93	Effects 3 Depth (Chorus Send Level)
37	LSB for Control 5 (Portamento Time)	94	Effects 4 Depth
38	LSB for Control 6 (Data Entry)	95	Effects 5 Depth
39	LSB for Control 7 (Channel Volume)	96	Data Increment (Data Entry +1)
40	LSB for Control 8 (Balance)	97	Data Decrement (Data Entry -1)
42	LSB for Control 10 (Pan)	98	Non-Registered Parameter Number (NRPN) - LSB
43	LSB for Control 11 (Expression Controller)	99	Non-Registered Parameter Number (NRPN) - MSB
44	LSB for Control 12 (Effect control 1)	100	Registered Parameter Number (RPN) - LSB
45	LSB for Control 13 (Effect control 2)	101	Registered Parameter Number (RPN) - MSB
48	LSB for Control 16 (General Purpose Controller 1)	120	[Channel Mode Message] All Sound Off
49	LSB for Control 17 (General Purpose Controller 2)	121	[Channel Mode Message] Reset All Controllers
50	LSB for Control 18 (General Purpose Controller 3)	122	[Channel Mode Message] Local Control On/Off
51	LSB for Control 19 (General Purpose Controller 4)	123	[Channel Mode Message] All Notes Off
64	Damper Pedal on/off (Sustain)	124	[Channel Mode Message] Omni Mode Off
65	Portamento On/Off	125	[Channel Mode Message] Omni Mode On
66	Sostenuto On/Off	126	[Channel Mode Message] Poly Mode On/Off
67	Soft Pedal On/Off	127	[Channel Mode Message] Poly Mode On
<b>digits feature name digits feature name</b>			
128	<i>Pitch Bend</i>	138	<i>Transpose by +/-8</i>
129	<i>Program Change</i>	139	<i>Velocity of all notes played</i>
130	<i>Channel After Touch</i>	140	<i>B4 chorus-vibrato</i>
131	<i>Standard keyboard action, glissando for pot</i>	141	<i>MidiTzer stops</i>
132	<i>Note-on</i>	142	<i>Ahlborn Archive stops</i>
133	<i>Note-off</i>	143	<i>Ahlborn Organs stops</i>
134	<i>Favorite Patches</i>	144	<i>Program Change select within a bank of 12</i>
135	<i>CC keyboard</i>	145	<i>Program Change Bank select</i>
136	<i>MIDI channel change for all</i>	149	<i>setup of default velocity for all notes</i>
137	<i>Transpose by +/-4</i>		