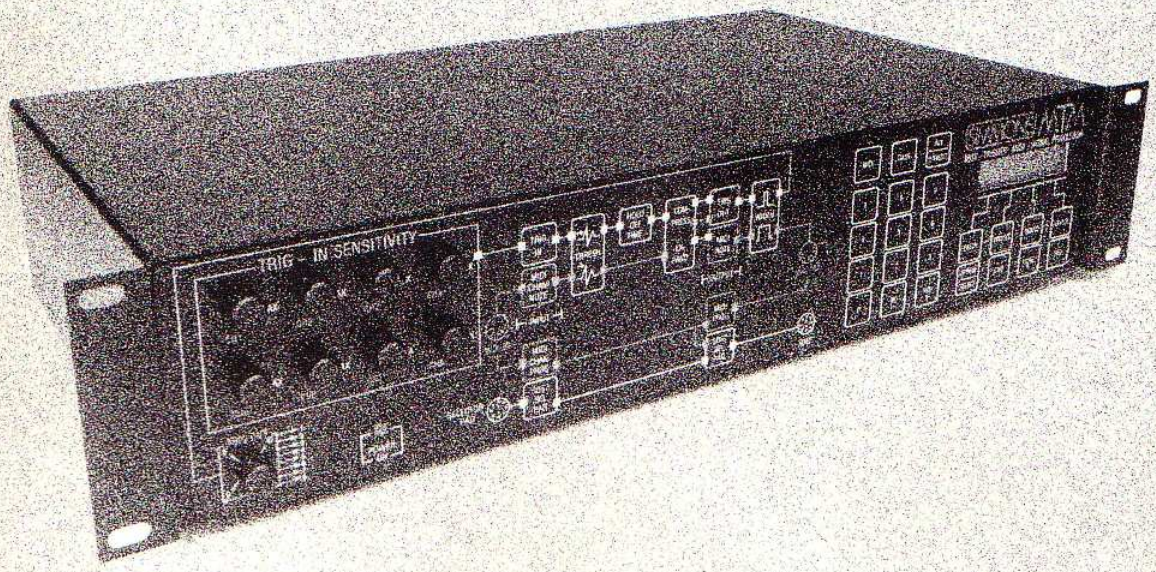


CREATIVE USE OF YOUR

MTM

SIMMONS

CONTENTS

ALL ABOUT THE MTM

MTM is an eight channel interface unit designed to translate audio signals into MIDI code and MIDI signals into triggers to drive other analogue synthesizers.

It can be used to interface SDS 7 to MIDI controlled synthesizers or sequencers and can be used to add such effects as compression or expansion of the pad dynamic range as well as adding repeat echoes or slap back echo to the SDS 7 and MIDI controlled sounds.

MTM will find many uses in the studio where acoustic sounds are required to trigger drum voices such as the SDS 7, SDS 9 or any other MIDI controlled voicing system. For example, a single hit on an acoustic tom tom can be the trigger of a whole sequence of notes and drum sounds.

MTM will also convert MIDI signals into analog triggers, thus allowing drum machines and sequencers easy access to the SDS 7 and other Simmons' synthesizers.

FEATURES

1. **TRIGGER PROCESSING** – MTM will clean up messy incoming audio signals and process threshold, compression, dynamic and hold off to produce clean trigger signals.
2. **OUT-PUT PROCESSING** – where pulse height and pulse width are under dynamic control.
3. Routing of channel inputs and channel outputs under program control.
4. Incoming triggers routeable to MIDI note and channel to play voices of other synthesizers.
5. Allows midi to change SDS 7 kit numbers via the selector pad.
6. A vast memory of more than 100 patches.
7. **SPECIAL EFFECTS** – programmable echo, MIDI note and chord layering which enable the player to pick different MIDI notes by how he strikes the drum. Programmable up and down chromatic scales, programmable run generator and a pitch generator.
8. Tape interface for storing data.
9. XLR and Jack inputs.
10. Dual footswitch option.
11. Full MIDI interface.
12. 50 factory setting to match common configurations.

MTM is housed in a 2U high standard 19 inch rack mount unit. Front panel consists of 35 membrane switches arranged in a graphic representation of their functionality. These switches are used for data entry and a 32 character liquid crystal display is used for data display. On the left hand side of the front panel are nine variable control knobs eight of which are used to control incoming audio levels and the ninth is used for overall output control levels so that inputs and outputs can be easily adjusted in a live situation. Inputs to the unit are located on the back panel and are via XLR type connectors as well as the standard ¼ inch jacks. Jacks are also used on the output and connect to existing Simmons' equipment via standard jack to jack cables either to the pad inputs in the case of the SDS 800 and SDS 8's or to the sequencer inputs for the SDS 7's and SDS 5's. Also located on the back panel are the selector pad's inputs and outputs to interface with the SDS 7, a footswitch input which enables the user to step forwards or backwards through program changes. Cassette input, output and obviously MIDI input, output as well as special channels assigned for SDS 7 hihat and stereo snare pad.

FUNCTIONAL DESCRIPTION

MTM's memory is divided into five areas, sequences, patches, processes, routes and effects. Each of these is split into factory/user areas.

PATCH
Each patch consists of a process number, a route number and an effect number, along with MIDI program change information and selector pad data for the SDS 7. MTM can store 20 factory and 99 user patches.

SEQUENCE
A sequence is a string of patch numbers. MTM can store 10 sequences of up to 100 patches long. These sequences can be stepped along with a footswitch.

PROCESS
There are 5 factory and 15 user processes. Each of these is a set of conditions for the inputs and outputs: the absolute threshold, hold off time, dynamic hold off, curve number, minimum output level, output fixed width, output dynamic width change for each trigger. Similarly for MIDI input signals there is adjustment for MIDI absolute threshold, MIDI compress/expansion curve and MIDI minimum output level, for MIDI out a fixed MIDI output width, and a dynamic MIDI output width.

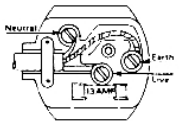
ROUTES
There are 20 factory and 79 user programmable routes, each route consists of relating inputs to outputs by trying together trigger in trigger out and MIDI note.

EFFECTS
These are 10 factory and 30 user effects such as Echo, run generator, sequences, chords, etc.

BEFORE YOU START

CONNECTING TO A MAINS SUPPLY
European mains voltage

Connect an appropriate mains plug to the mains cable according to the following colour code.



Brown – Live
Blue – Neutral
Green/Yellow – Earth (Ground)

Check that the voltage label on the back of the panel matches your domestic mains supply.

240v – G.B. and Australia
220v – Europe
115v – U.S.A. and Canada
110v – Japan

MTM is a computer-controlled processor and should be treated with care. A few simple rules, if followed, will avoid problems in the future.

They are:

Try and use a clean power source, away from equipment that may produce transient spikes through the mains power, i.e. electric motors, heavy switch gear etc.

MTM is supplied with a three core power cord – use this with a grounded AC power source.

Do not place MTM on top of speaker cabinets or amplifiers which might subject it to excessive heat and vibration.

FACTORY SETTING

We realise that some of the concepts for programming MTM may be difficult to understand or that applications may not immediately be apparent. Therefore there are a number of pre-set factory processes which should suit most applications for MTM. In most applications they would only need 'tweeking' to work as desired. They are:-

1. Mono Jack Pads
2. Mono XLR Pads
3. Stereo Jack Pads (Quatro set)
4. Kit set up-Jack pads

MTM CHANNELS

- 1 Bass
- 2 Tom
- 3 Snare } STEREO PAD
- 4 Rim } STEREO PAD
- 5-8 Toms

5 Acoustic drums

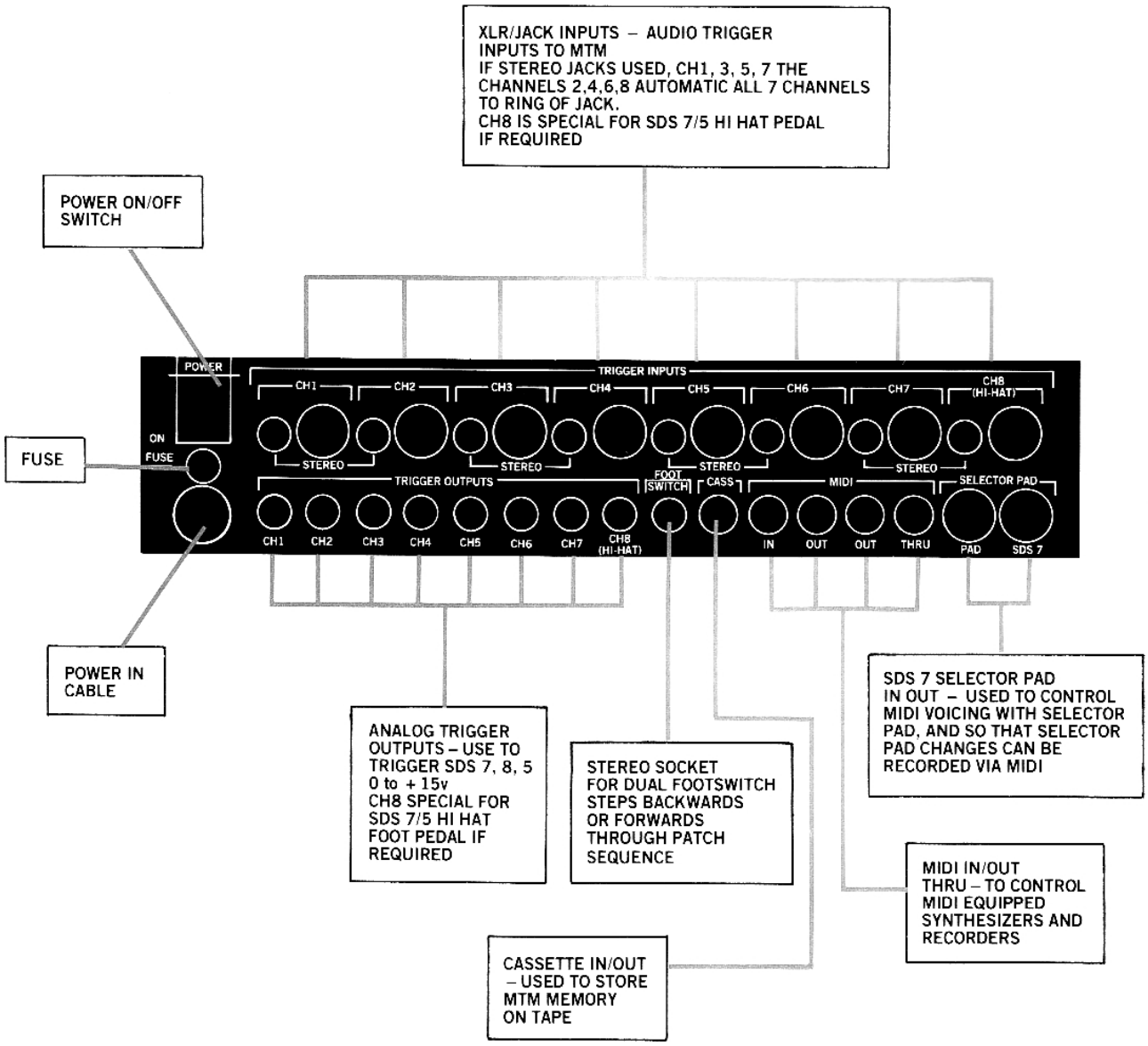
MTM's factory patches have been set up to give a demonstration of effects and routes as applied to MIDI. This demonstration can only be effective if MTM is plugged into some sort of MIDI controlled voice generator, where it will demonstrate repeat echoes, slap back, sequences, chord layering, glissandos etc.

PATCH NO.	TRIGGERING DEVICE	USES			TYPE OF EFFECT	SELECTOR PAD	MIDI CH/PROG
		PROCESS	ROUTE	EFFECT			
1.	Mono Jack Pads	1	1	1	Normal (no effects)		1/1
2.	Mono Jack Pads	1	2	2	Echo		1/2
3.	Mono Jack Pads	1	3	3	Chord		1/3
4.	Mono Jack Pads	1	4	4	Mixture & Sequence		1/4
5.	Normal XLR Pads	2	1	1	Normal (no effects)		1/1
6.	Normal XLR Pads	2	2	2	Echo		1/2
7.	Normal XLR Pads	2	3	3	Chord		1/3
8.	Normal XLR Pads	2	4	4	Mixture & Sequence		1/4
9.	Stereo Jack Pads	3	1	1	Normal (no effects)		1/1
10.	Stereo Jack Pads	3	2	2	Echo		1/2
11.	Stereo Jack Pads	3	3	3	Chord		1/3
12.	Stereo Jack Pads	3	4	4	Mixture & Sequence		1/4
13.	Kit Set up-Mono Jack	4	1	1	Normal (no effects)		1/1
14.	Kit Set up-Mono Jack	4	2	2	Echo		1/2
15.	Kit Set up-Mono Jack	4	3	3	Chord		1/3
16.	Kit Set up-Mono Jack	4	4	4	Mixture & Sequence	↓	1/4
17.	Acoustic Drums	5	1	1	Normal (no effects)	–	1/1
18.	Acoustic Drums	5	2	2	Echo	–	1/2
19.	Acoustic Drums	5	3	3	Chord	–	1/3
20.	Acoustic Drums	5	4	4	Mixture & Sequence	–	1/4

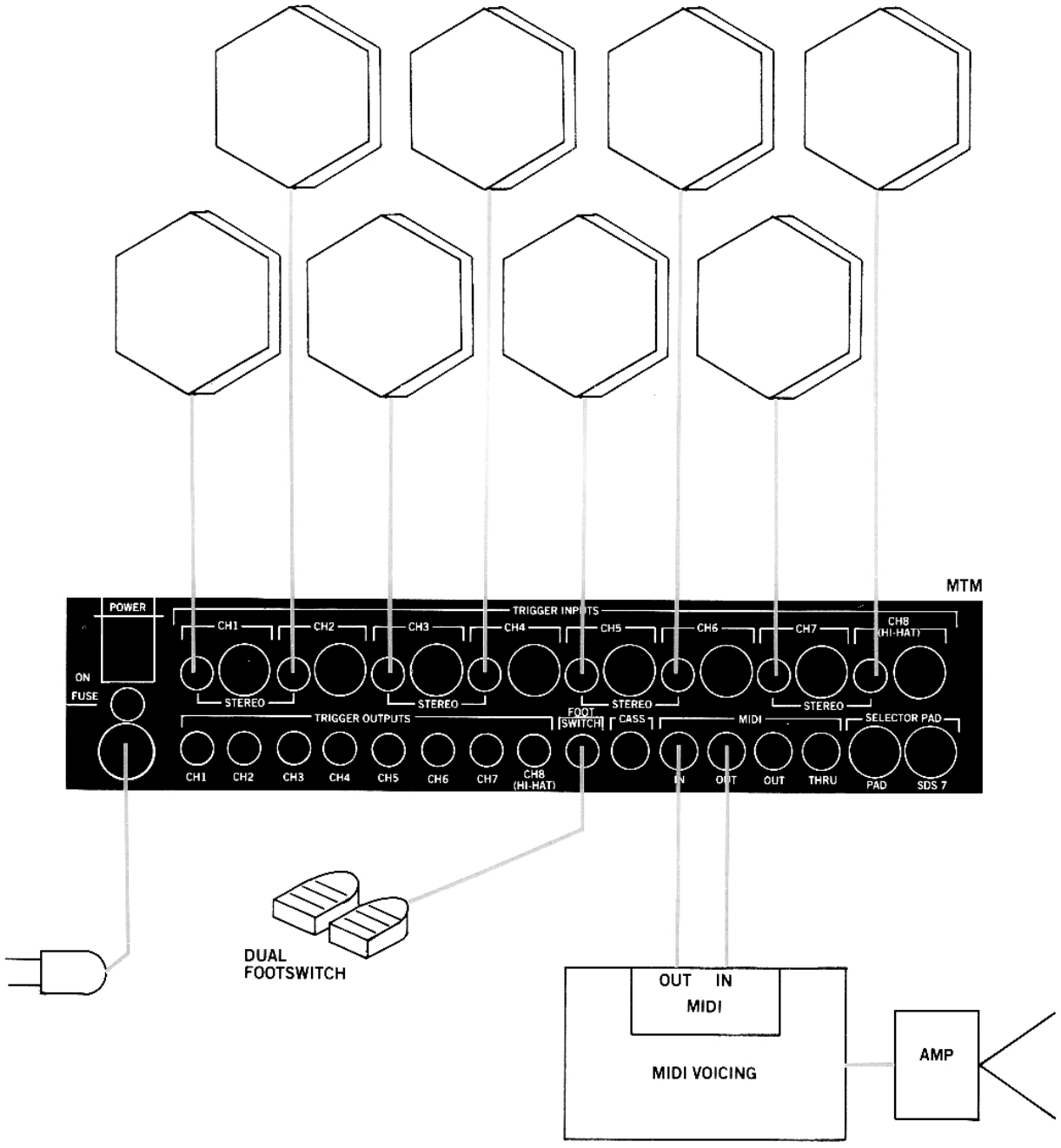
You probably know enough now to start using MTM in this basic form. Let's connect it up.

Refer to the following diagrams for your particular set up.

CONNECTING UP MTM BACK PANEL

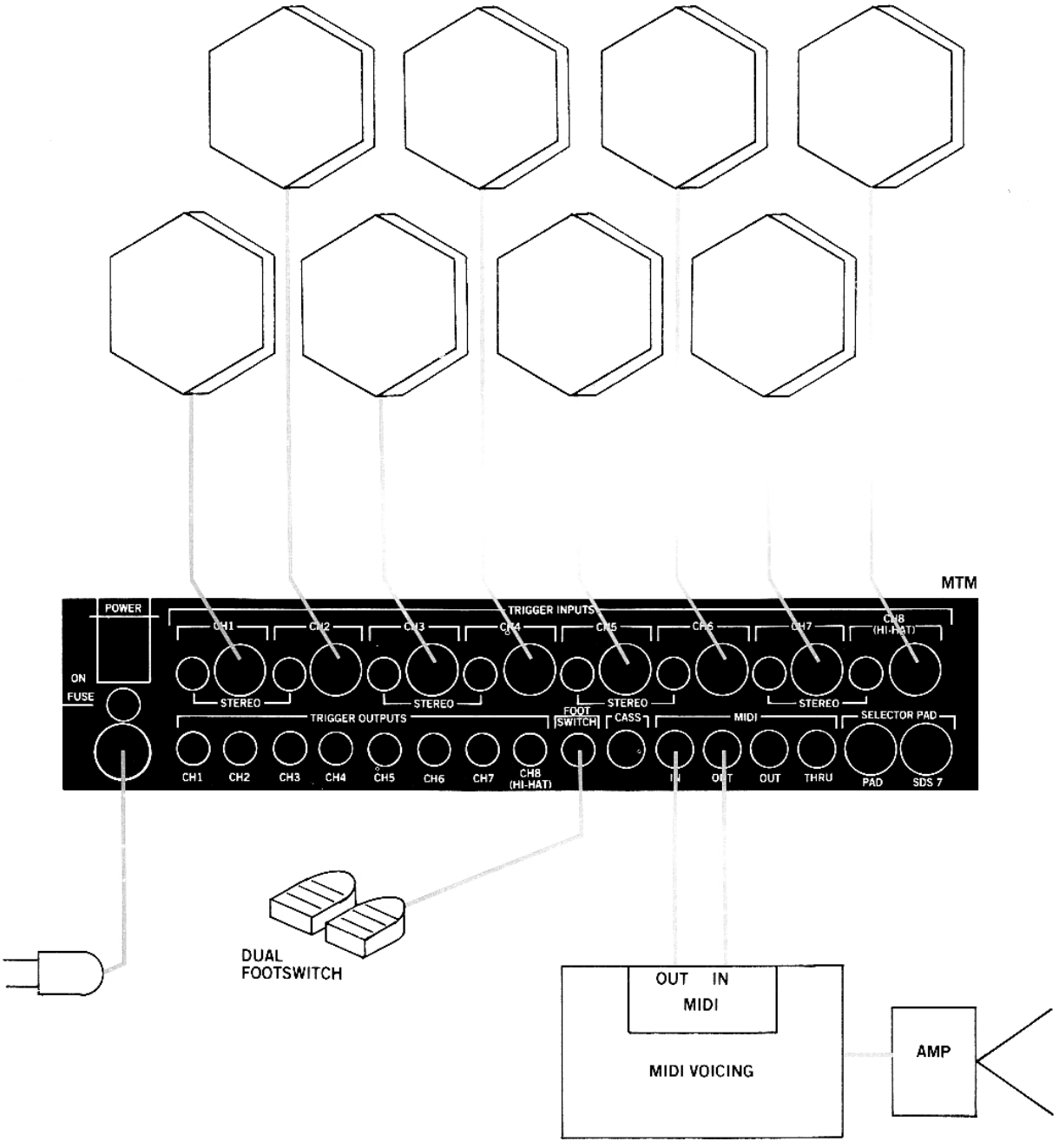


MTM AND THE MONO JACK PAD SET UP PATCHES 1-4



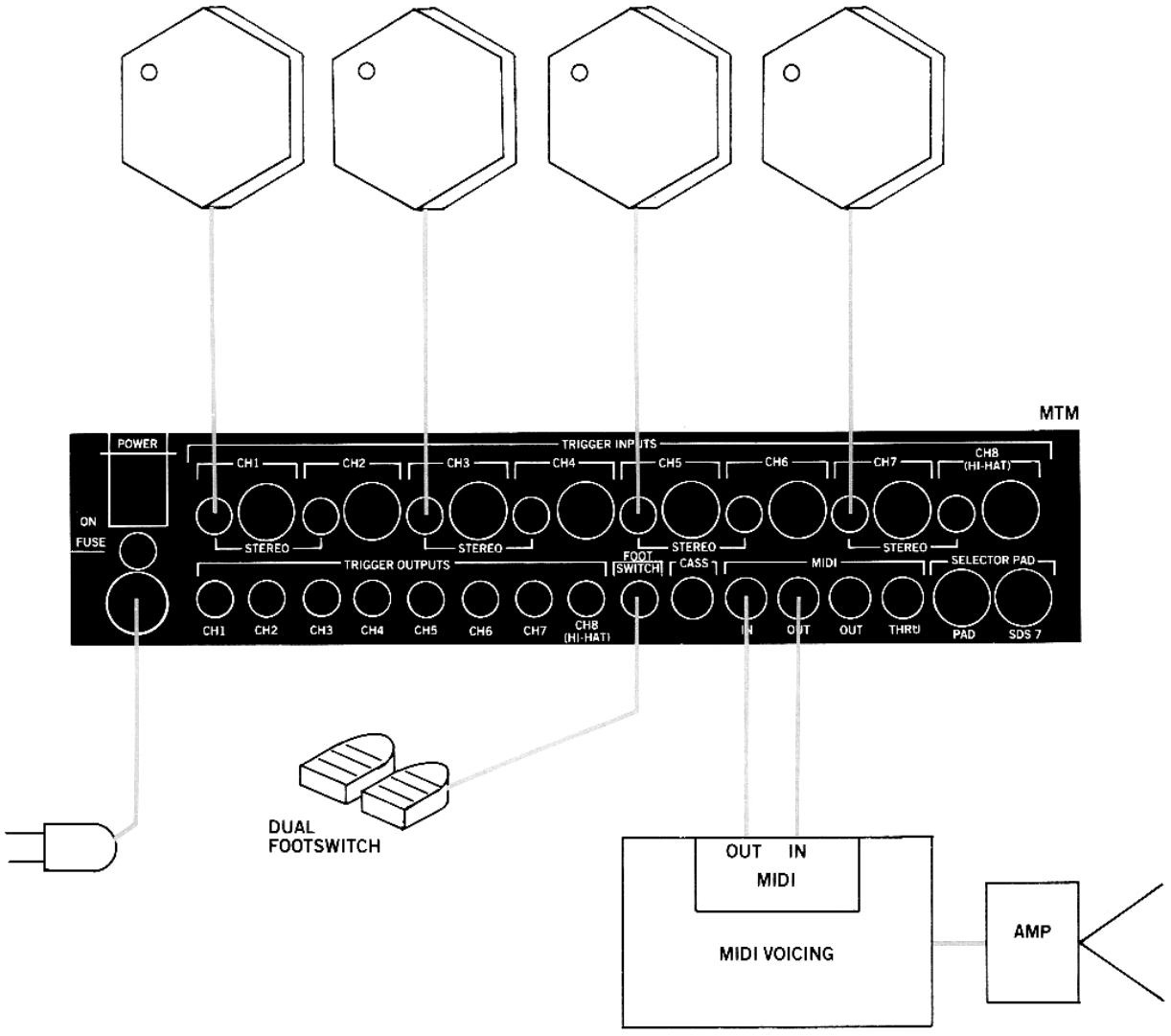
MIDI voicing can be anything with a good MIDI interface. Footswitch will step MTM and MIDI voicing for production of many percussion sounds.

MTM AND THE MONO XLR PAD SET UP PATCHES 5-8



MIDI voicing can be anything with a good MIDI interface. Footswitch will step MTM and MIDI voicing for production of many percussion sounds.

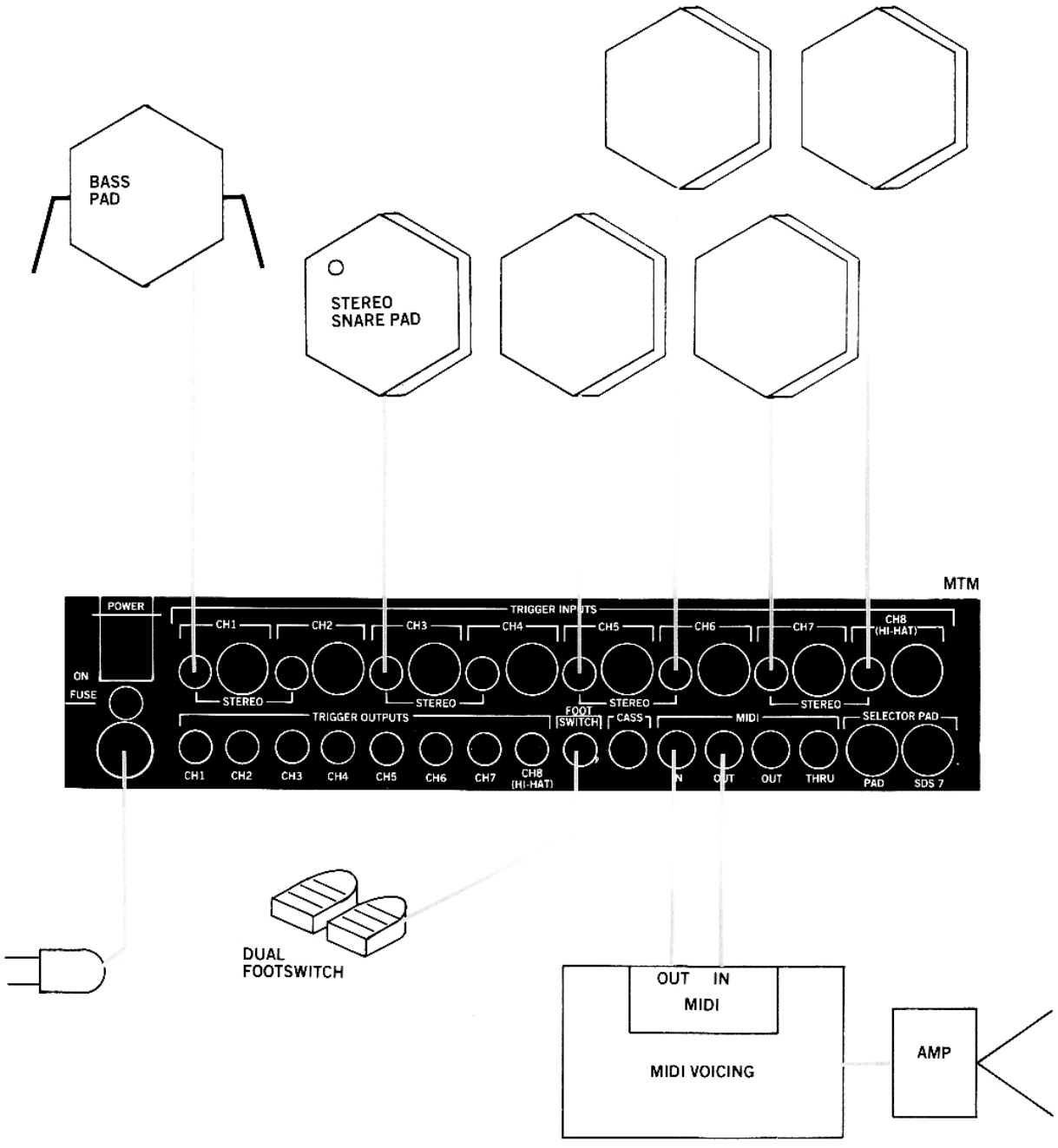
MTM AND THE STEREO JACK PAD SET UP PATCHES 9-12



Stereo pads plug into channels 1,3,5,7, to play all 8 MTM channels. Pad surfaces play 1,3,5,7, pad rims play channels 2,4,6,8.

MIDI voicing can be anything with a good MIDI interface. Footswitch will step MTM and MIDI voicing for production of many percussion sounds.

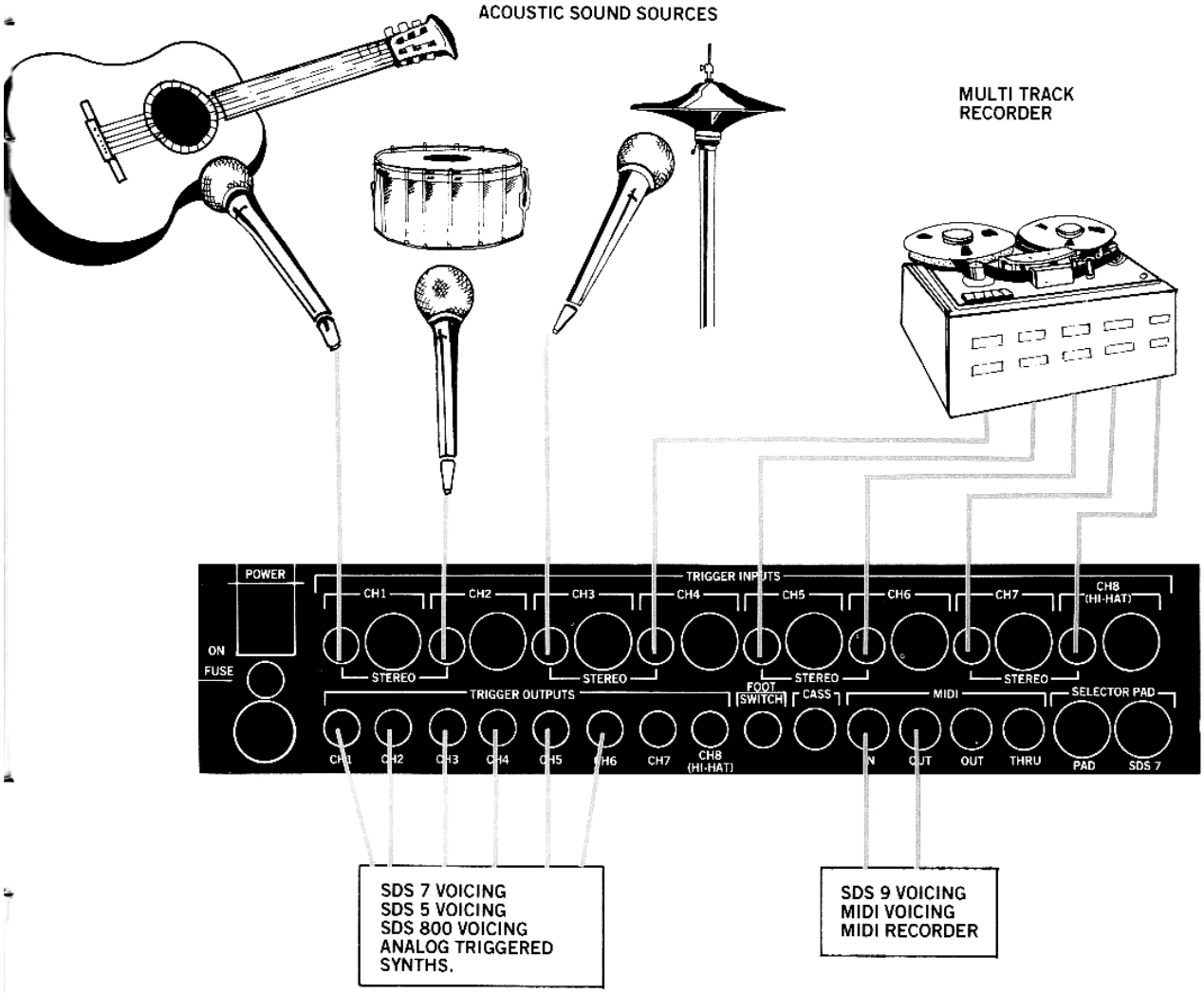
MTM AND THE KIT SET UP
PATCHES 13-16



Bass Pad uses CH1.
Stereo Snare Pad uses CH3 and CH4.
CH2, 5, 6, 7, 8 are Mono Pads.

MIDI voicing can be anything with a good MIDI interface. Footswitch will step MTM and MIDI voicing for production of many percussion sounds.

MTM USING ACOUSTIC SOURCES TO TRIGGER ANALOG SYNTHS AND MIDI EQUIPMENT – PATCHES 17-20



POWERING UP

Switch MTM on by turning on the main switch at the back of the unit.

The red POWER ON led indicator on the left of the panel will light up and the LCD (liquid crystal display) on the right of the unit will display SIMMONS MTM(c) 1985 v01.

The v01 is the software version installed in your unit, other versions of the software may be installed at later dates. After a few seconds the display will change to:

play patch			F01
F01	01	01	01

This means that MTM is in the 'play (back)' mode and is 'playing' factory patch number 1, which consists of process1, route1 and effect1.

To the left of the LCD are two buttons labelled − 1.NO. , and + 1.YES.>. These two buttons are used to change values throughout the use of MTM. They are referred to as (+) (−)

At the moment they will advance (+) or retard (−) the patch number being displayed in the LCD. Single presses of the buttons change the display by single numbers, if you hold the button down, the numbers will change automatically, slowly at first but speeding up as you continue to press the button. This is so you can quickly "scan" from the highest value to the lowest without having to continually stab the buttons. When the number in the display reaches its highest value, it starts again from the lowest (unless there is an 'off' value i.e. 0, in which case it stops), in this case patch numbers will cycle 98.99.1.2.3.4 etc. going up and 4.3.2.1.99.98.97 coming down.

Choose the patches that are relevant to your set up and cycle backwards and forwards around those patches. Strange things may happen if you stray from the appropriate patches, eg. signals from acoustic drums will send MTM berserk if its expecting the relatively clean signals produced by Simmons' pads.

NOTE: As you change patches you will notice that different process, route and effect numbers are displayed and therefore used. It is the process contained in the patch that matches the incoming trigger signals. These have been tailored to suit the configuration.

ALTERING MTM PATCHES

To save confusion this manual assumed the use of 8 mono jack Simmons' pads. If using other configurations, the examples and numbers displayed will be different.

An MTM 'patch' consists of three parts, the process, route, and effect. During patch playback any one of these three parts can be substituted to alter the patch. For example if you are using patch 1, and you want to hear what that sounds like with slow echo added. You know that effect number 2 is programmed to be just that.

The display is now telling you that MTM is playing patch 1, which consists of process 1, route 1, and effect 1.

Pressing EFFECT changes the display as follows:

play effect			01
F01	01	01	01

You can now alter the effect number using the −/+ buttons as before. Press the + button until effect 2 is displayed.

play effect			02
F01	01	01	02

The display is now telling you that MTM is playing patch1, which consists of process1, route1, and effect 2.

If you now hit a pad you will hear the effect no. 2 (slow echo).

You can make similar alterations to the other two parts of the patch. – press PROCESS, the display will change:

play process			01
F01	01	01	02

Once again change the number by using the −/+ buttons, say to 15.

The display will look thus:

play process			15
F01	15	01	02

Similarly to change the route, press ROUTE, change the number with −/+ say to 7, the display will look like this:

play route			07
F01	15	07	02

Which is telling you that you are listening to a patch consisting of PROCESS 15, ROUTE 7, and EFFECT 19.

BE CAREFUL. THIS IS NOT PATCH 1!! Patch one is still process 1, route1, and effect1, the same as when you started – you have only altered patch 1 temporarily for experimentation. If you want to save this new patch permanently, you have to decide on a patch number and STORE it away under that new number (it could be patch 1 if you want) and the old patch residing at that number is over written with the new patch.

Changing between factory and user patches

To date you have been using factory patches (denoted by the letter 'F' before the patch No.). to alternate between factory and user press the 'All/Default' button. The 'F' will disappear showing that you are in user.

STORING PATCHES

At this point you can try storing your experimental patch.
Press STORE.

The display will change:

store patch			F01
	15	7	02

Press store again to actually store the patch at this patch number.

The display will tell you:
Can't store patch.

This is because you are attempting to store the edited patch F01 back into factory F01 which is a protected area and cannot be changed. You must go to the user area to store your edit.

Press store, select your user patch number by pressing All/Default to enter user area.

The 'F' in front of the patch number disappears – you are now in the user area.

You can now decide to at which patch number you want to store this particular patch. Change the patch number with the –/+ buttons, to 99. (did you go up to 99 through 96.97.98. etc. or did you go backwards?). The display should read:

store patch			99
	15	7	02

Now to finally store the patch away, press STORE. (You can press any alternative button at this stage which will abort the 'store' process and return you to the previous level.) The display will momentaraly show – STORING PATCH 99, and then return to the play patch stage displaying:

play patch			99
	15	7	02

Now you can change the three elements that go to make up an MTM patch, swap them around to see what their effect is, and store the new combinations as new patches.

You can use the factory patches and modify them to suit your own requirements, mixing the effects, processes and routes to form new patches. Sooner or later you will want to modify some of the hundreds of individual parameters that go to make up a route, a process, or an effect.

Read on.

Factory Settings – For your information

Patches – As per table on page 5

Processes

1. – Mono Jack pads.
2. – Mono XLR pads.
3. – Stereo pads.
4. – Kit set-up.
5. – Acoustic drums.

Routes – MIDI Data (all transmitted and received on MIDI CH1).

	CH1	CH2	CH3	CH4	CH5	CH6	CH7	CH8
1	C4	E4	G4	B4	C5	E5	G5	B5
2	C4	E4	F4 ♯	B4	C5	E5	F5 ♯	B5
3	C4	E4	F4 ♯	A4	C5	E5	F5 ♯	A5
4	C4	D4 ♯	F4 ♯	A4	C5	D5 ♯	F5 ♯	A5
5	C5	C5 ♯	D5	D5 ♯	E5	F5	F2	C5
6	G6	A6	A6 ♯	C7	D7	E7	F7	G7
7	A3	G4	A4	C5 ♯	E5	A5	C6 ♯	E6
8	A5	C6	D6 ♯	A6	C7	D7 ♯	A7	C8
9	A3	D4	A4	C5	D5	G5 ♯	C6	D6
10	A4	C5	G5 ♯	D5	E5	G5	A5	C6
11	C5	E5	G5	C6	E6	G6	A♯ 7	C7
12	C5	D♯ 5	G5	C6	D♯ 6	G6	A♯ 6	C7
13	D5	F♯ 5	A5	D6	F♯ 6	A6	C7	D7
14	D5	F5	A5	D6	F6	A6	C7	D7
15	E5	G♯ 5	B5	E6	G♯ 6	B6	D7	E7
16	E5	G5	B5	E6	G6	B6	D7	E7
17	F5	A5	C6	F6	A6	C7	F7	A7
18	F5	G♯ 5	C6	F6	G♯ 6	C7	F7	A7
19	C6	A7	D6	C5	D5	C6	C6	D6
20	G6	E8	A6	G5	A5	G6	G6	A6

ROUTES

Triggers
All routes 1-9 are connected 'through' (ie. In 1 to out 1, in 2 to out 2) etc.

Route 20 has ins to outs mixed up for your enjoyment, demonstrating MTM's trigger routing capability – try swapping bass and snare drums etc.

Effects

1. – No effects.
2. – Slow echo.
3. – Chords.
4. – Mixture of chords, echoes, real time sequence.
5. – Echo, MIDI 'up in ones' – each channel different speed.
6. – Echo, MIDI 'down in ones' – each channel different speed.
7. – Echo stepped sequence – each channel different speed.
8. – Trigger stepped sequence. CH1-4 auto stepped CH5-8.
9. – Splits.
10. – Layers.

NOTE: Some of the effects come in at hard dynamics, others progressively as you strike the drum harder.

Try experimenting with these different factory routes and effects.

At this stage you should be able to swap processes, routes, and effects, and store them away in MTM's memory. The next step is to edit the patch itself.

PATCH EDIT

You know that a patch consists of 3 items – A process number, a route number and an effect number. There are two other pieces of information which are part of a patch.

These are the MIDI patch change data and the SDS 7 selector data.
This is the entire patch data:

Patch = process, route, effect, MIDI chan, MIDI prog, SDS 7 selector number.

To edit a patch, press patch to enter playback patch and then press edit to enter edit patch.

The display will change.

Patch Edit		F01
01	01	01

You can press process, route or effect, the cursor will move under the appropriate numbers. Use +/– to change the data if required.

To edit the other items contained in a patch (MIDI chan, MIDI prog and selector pad) press the MIDI chan/Prog, the MIDI prog, or selector buttons. (The lower group of buttons).
To finish edit patch, press the edit button to lose what you have changed, or, press the store button to save the new patch.

More detailed descriptions follow:-

MTM AND MIDI PROGRAM CHANGE DATA

Every time you change a patch on MTM, MIDI can instruct an external MIDI controlled instrument to change patch as well. So, for example if MTM was being used as an interface between Simmons' pads and a synthesizer, every new MTM patch number could be a different sound. Patch 1 plays tubular bells, patch 2 woodblocks, patch 3 cowbells etc. Every time you choose a new patch on MTM a signal is sent to the slave synthesizer that says 'change to sound x'. These different sounds are normally called programs or patches, so you have MTM's patch number calling up the slave synthesizers' program or patch number.

To send data down MIDI you have to choose a 'MIDI CHANNEL'. The receiving instrument has to be set on this channel as well. MIDI can send information down one of 16 'channels', these channels are numbered 1 to 16. If MTM was sending MIDI down channel 1, but the slave synthesizer was 'listening' on channel 2 obviously they will not be able to communicate.

Each MTM patch can send MIDI down a different 'channel', so that different MTM patches can be routed to different slave instruments. (E.g. a DX7 could be listening on channel 1, whilst a Jupiter 8 could be listening on channel 2. Then MTM could be programmed to send patches 1 to 10 on channel 1, and 11 to 20 on channel 2. – A very convenient way to control many slave synthesizers without having to swap wires and cables.) A more detailed description of MIDI is given later in this manual.

Once you have decided which MIDI channel each MTM patch is sending on, you can choose which program change number MTM sends.

This means that you have flexibility over the selection of external voicing – Patch 1 on MTM can call up program 45 on a DX7, patch 2 on MTM Can call up program 12 on a Jupiter 8.

PROGRAMMING MIDI CHANNEL SELECT AND PROGRAM CHANGE. As described above, you have to set up two numbers for each MTM patch. MIDI channel (1 to 16), and MIDI program change (1 to 128).

This is achieved in edit patch.

Press PATCH then EDIT, then press (+)/(–) to change the patch number if required.

For demonstration select patch 01 and then press MIDI CHAN/PROG. The display will show:

patch edit	01
Chan:01	Prog:001

Which is telling you that MTM patch 01 is transmitting MIDI program change information on MIDI channel 1.

To change the MIDI channel press the (+)/(–) buttons. You will be able to change to any of the 16 MIDI channels. When you select past MIDI channel 16 you will see the message:

Chan:- = prog: – – –

This means that MIDI program change data will not be transmitted at all for that patch number.

Press the + button once again to select MIDI Chan 01.

This means that MIDI patch change data will be sent on MIDI channel No. 1.

To change the MIDI program change number press OPTION, or MIDI prog buttons.

patch edit	01
Chan:01	prog:001

Which is telling you that MTM patch 1 will send program change 001. This means that whenever you select patch 1 on MTM, MTM will tell the slave synthesizer to change to program 1 as well.

To alter the 'program change' number press (+)/(–).

For example:

patch edit	01
Chan:01	prog:024

This means that when you select patch 1 on MTM the slave synthesizer will change to program number 24.

Remember to store any changes by pressing STORE, STORE.

SDS 7 SELECTOR PAD

THE SDS 7 selector pad normally plugs directly into the SDS 7, and is used to select one of sixteen kit numbers on the SDS 7. The selector pad itself is split up into sixteen sections numbered 1 to 16.

It would be useful if the SDS 7 selector pad could not only change the SDS 7's kit number, but also change the program number of an external 'slave' synthesizer. To do this MTM must know when the selector pad has been pressed.

There are two special sockets on MTM which allows it to be 'inserted' between the selector pad and the SDS 7. The sockets are mounted on the back of MTM and are labelled PAD and SDS 7. To utilise this feature you just buy the MTM accessory pack which includes the special lead that connects MTM to the SDS 7 selector pad input.

Connect the selector pad to the socket labelled PAD, and the SDS 7 to the socket labelled SDS 7.

When you press the selector pad MTM then tells the SDS 7 to change kit number, as well as sending the relevant 'program change' date down MIDI to an external MIDI controlled 'slave' synthesizer.

The selector pad can only select MTM patches 1 to 16, and MTM can only select one of sixteen SDS 7 kits. (Those that have been programmed on the SDS 7 selector pad).

NOTE: The selector pad only works in playback patch mode. It is disabled in all other modes.

EDITING THE SDS 7 SELECTOR PAD

Pressing any of the sixteen sections on the selector pad will always change MTM to that patch number. I.E. pressing selector number 1 will always select patch 1 on MTM, selector 2 patch 2, selector 3 patch 3 etc.

MTM then passes this instruction on to the SDS 7 without altering it. As far as the SDS 7 is concerned it is connected directly to the selector pad.

This is fixed on MTM and cannot be changed. You program the SDS 7 as usual to select which selector pad section selects which kit (see the SDS 7 manual), and MTM to select which program change number is sent down MIDI to the slave synthesizer.

To program MTM's program change number for patches 1 to 16 refer to MTM AND MIDI PROGRAM CHANGE DATA. The patch change data entered for patches 1 to 16 as described in this chapter will be sent when you press the SDS 7's selector pad.

MTM patches numbered 17 to 99 can also select one of the sixteen kits on the SDS 7.
Enter edit patch and change to MTM patch 4.
Press SDS 7 SEL PAD located on the INPUT side of MTM.
Note that the display will show:

patch edit	04
selector in	:04

This is telling you that the SDS 7 selector pad 4, will always select MTM patch 4. You cannot change this number, selector pad 1 to 16 is always tied to MTM patch 1 to 16. Return to patch edit by pressing PATCH. Using the (+)/(−) buttons select patch number 17.
Press SDS 7 SEL PAD on the output section of MTM, the display will show:

patch edit	17
selector SDS 7:	01

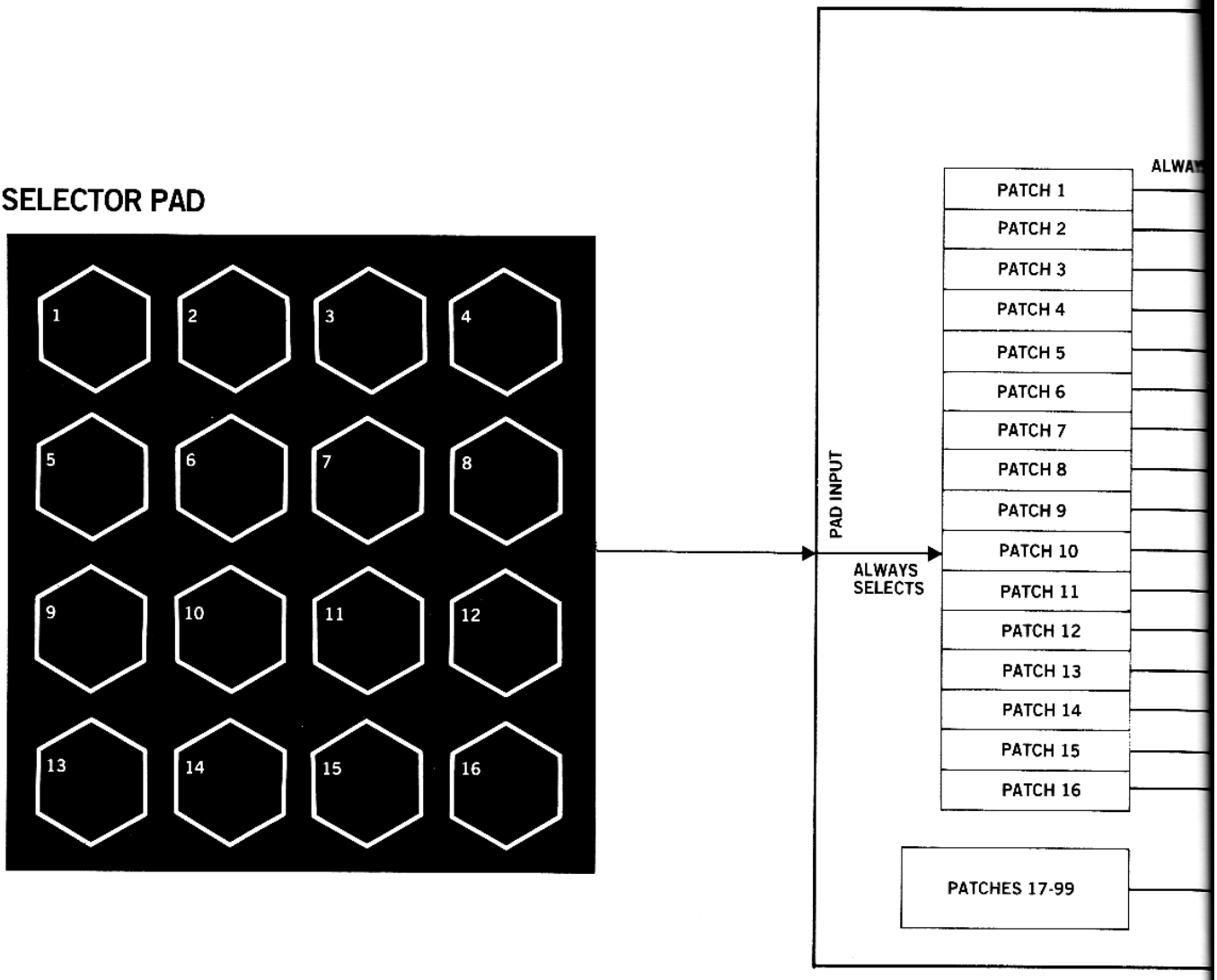
Press the (+) button to change the selector number.
Note that you can only select numbers between 1 and 16 or off (denoted by − − in the display.)
For example you program it to be 10, the display will look thus:

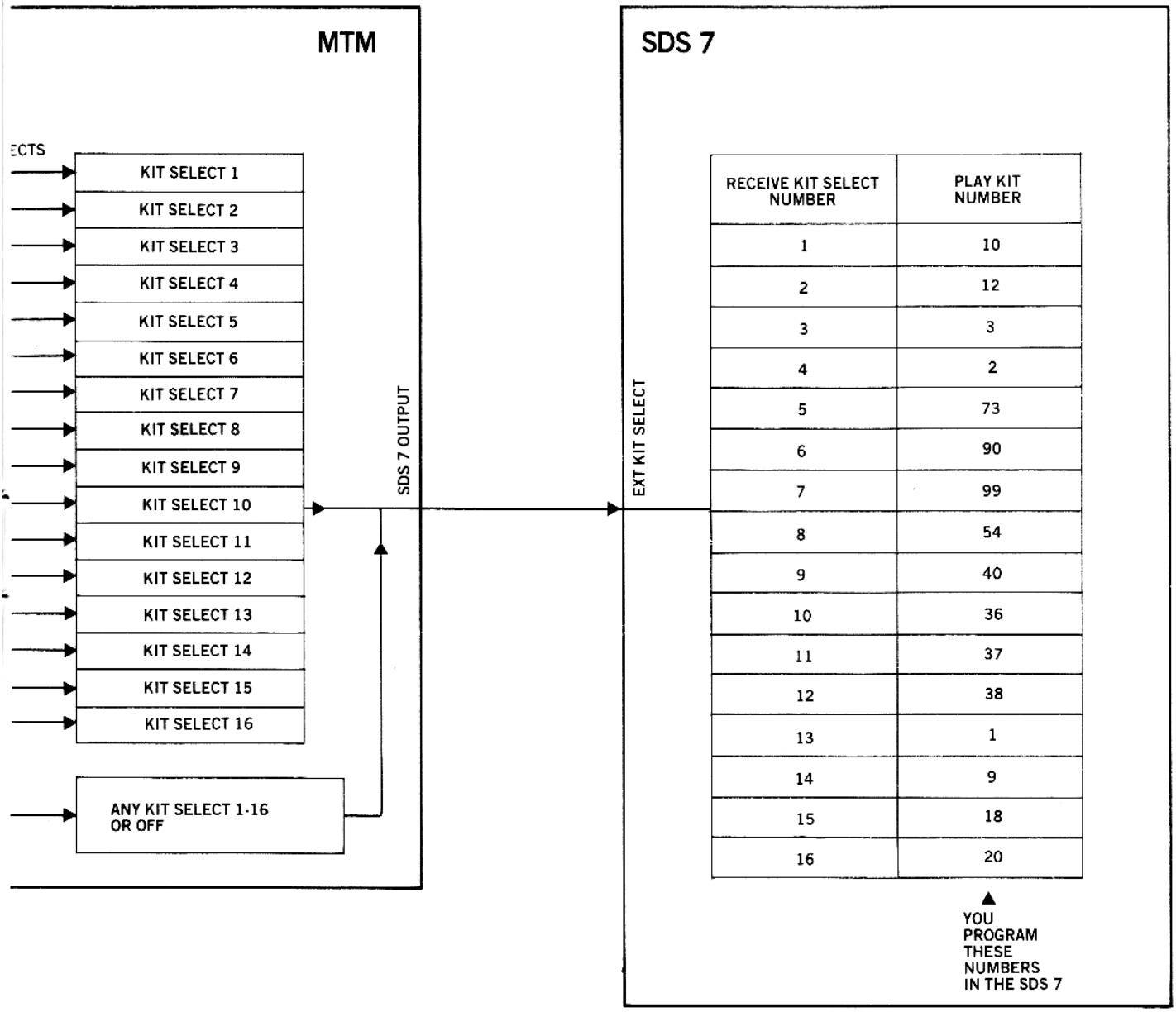
patch edit	17
selector SDS 7:	10

This would mean that every time you select patch number 17 on MTM, the SDS 7 would think that you had pressed selector 10 and change to the appropriate kit number.

To store the edit don't forget to press STORE, STORE.

SDS 7 SELECTOR PAD/MTM/SDS 7




MTM
ECTS
KIT SELECT 1
KIT SELECT 2
KIT SELECT 3
KIT SELECT 4
KIT SELECT 5
KIT SELECT 6
KIT SELECT 7
KIT SELECT 8
KIT SELECT 9
KIT SELECT 10
KIT SELECT 11
KIT SELECT 12
KIT SELECT 13
KIT SELECT 14
KIT SELECT 15
KIT SELECT 16
ANY KIT SELECT 1-16 OR OFF
SDS 7 OUTPUT
EXT KIT SELECT
SDS 7
RECEIVE KIT SELECT NUMBER
PLAY KIT NUMBER
1 10
2 12
3 3
4 2
5 73
6 90
7 99
8 54
9 40
10 36
11 37
12 38
13 1
14 9
15 18
16 20
YOU PROGRAM THESE NUMBERS IN THE SDS 7

SEQUENCE OF PATCHES

When you have programmed your own patches into MTM, or wish to use the 'factory' patches, it is useful in a 'live' situation to sequence the desired patches together in the order in which they are going to be used.

You can then re-call that particular sequence and step through it using the (+) button on the front panel, or the footswitch that connects to the back of MTM. You can also step backwards through the sequence using a second footswitch, or the (−) button.

A sequence is a 'string' of patch numbers.

Each sequence has a number. MTM can store 10 of these sequences, and each sequence can be up to 99 patches long.

Displaying a sequence (of patches)
From the 'switch-on' state, with 'play patch 01' being displayed press SEQUENCE.

The display will look thus:

playseq:0

To select a different sequence, use the keypad buttons 0-9. Obviously button 1 will select sequence 1, button 2 sequence 2 etc.

Select sequence 0.
Press (+).

display:

playseq:0			pos:00
F01	01	01	01
Patch	Process	Route	Effect

The display is now telling you that sequence number 0, position 00 is assigned to MTM patch 01, (which consists of process 01, route 01, and effect 01.)

To advance through the sequence press (+).
Each press of (+), brings up the next patch that has been stored in sequence 0. I.E. the patch number stored in position 01, 02, 03, . . . 98. 99. along with what that patch consists of.
Press (−) to step backwards through the sequence.

So, a sequence of patches can be visualised:

Sequence number N (0-9)

Position (0-99)	patch number (1-99)	= proc	+ route	+ eff
00	NN	NN	NN	NN
01	NN	NN	NN	NN
02	NN	NN	NN	NN
99	NN	NN	NN	NN

The maximum number of positions in each sequence is 99, but you do not have to use them all. When you program a sequence you can define the end of the sequence. At this point if you continue to press the (+) button the sequence will start again from the beginning.

A typical sequence would look like this:

Sequence number 6

Position	patch number	= process	+ route	+ effect
00	20	01	01	15
01	15	15	12	15
02	01	01	01	01
03	02	02	02	02
04	15	15	12	15
05	99	01	02	01
end				

You can then cycle around these patch numbers during a performance just using the (+)/(−) buttons or the forward/backward double footswitch.

EDITING THE SEQUENCES

You can edit an existing sequence by **overwriting, inserting,** or **deleting** the patch numbers for the position chosen.

From **playseq** press EDIT.

the display will show:

seqedit:0			pos:00
F01	01	01	01

Which is telling you that you are in sequence edit mode, editing sequence number 0, position 00 and that the patch assigned to that position is patch 01, (consisting of process 01, route 01, effect 01.)

Press OPTION.

display:

```
seqedit:0                    pos:00
F01                          overwrite
```

If you want to overwrite position 00 with a new patch number press enter the new patch number using the keypad.

The new patch number will appear to the right of the existing patch number, and then move left and replace it.

```
seq edit:0                   pos:00
01              45           overwrite
```

```
seqedit:0                    pos:00
45                           overwrite
```

If you want to overwrite position 2, press (+) to move onto the next position and then enter the new patch number as required using the keypad.

Remember to STORE the updates by pressing STORE, STORE, if you want to save the new sequence.
You can step back through the sequence to overwrite a position using the (–) button.

INSERT
To insert a patch into an existing sequence (i.e. make the sequence longer), press OPTION.

The display will change to:

```
seqedit:0                    pos:02
01                           insert
```

This is telling you that at position 02 is patch 01, and that you can now insert a new patch after position 02, in position 3. The patch that was in position 3 is moved to position 4, the patch that was in position 4 is moved to 5 etc.

Example: sequence 6.

pos	00	01	02		03	04	05
patch	20	15	01	insert	02	15	99

To insert the new patch simply enter the new patch number using the keypad. (Again always enter a two digit number like 09 not 9.)

The display will then show the new patch number at its new position.

```
seqedit:6                    pos:03
88                           insert
```

Patch 88 has been inserted at position 3, and the sequence has been extended by 1.

pos	00	01	02	03	04	05	06
patch	20	15	01	88	02	15	99

You can insert as many patches as you like, each new patch will be assigned a new position number, and the rest of the sequence is moved right.
If you see the message SEQUENCE IS FULL then the sequence has 99 patches already and this is the maximum MTM allows for any sequence.

DELETE
The third option in edit sequence is DELETE.
Press OPTION to select delete.

display:

```
seqedit:6                    pos:01
15                           delete
```

The display is telling you that you are editing sequence 6, position 01, and that the patch stored at that location is patch number 15 and that you can delete that patch from the sequence if you require.

To delete the patch simply press 0 in the keypad. The patch currently stored at position 01 is deleted, the remaining patches in the sequence move left to fill the space, and the sequence is reduced in length by one.

You can carry on deleting as many patches as you like simply by pressing the '0' key.

You will see the message CANT MAKE EMPTY if you try and delete the last patch in the sequence, as MTM will not allow you to have a sequence of patches without any patches in them.

FACTORY – USER

To swap between factory patch number and user numbers, select either INSERT or OVERWRITE modes and press the ALL/DEFAULT button.

INITIALISING THE SEQUENCE

To clear out a sequence, press ALL/DEFAULT while in delete mode. This will ask you if you actually want to clear the sequence, press (YES) if you want to. Sequences are initialised to being just 1 patch – the first factory patch, F01.

Remember once again any changes you have made to a sequence are stored only when you press STORE (select seq no.) STORE. Display will show STORING SEQUENCE NN when it is making a permanent copy of the edited sequence.

SUMMARY OF EDIT SEQUENCE

To play a sequence

1. Press SEQ to enter **play sequence.**
2. Press 0 to 9 in keypad to select **sequence number.**
3. Press (+)/(–) to step **forward** or **backwards** through the sequence, (or use the footswitches.)
4. Press SEQ to return to 2.

To edit a sequence

5. Press EDIT.
6. Press OPTION to select overwrite.
   insert
   delete

If overwrite

7. Press (+)/(–) to move to desired position.
8. Press 0 to 9 in keypad to overwrite data, (has to be a two digit number.)
9. Press ALL/DEFAULT to toggle factory/user patch.
10. Press STORE, STORE if you want to save the edit.
11. Press EDIT to return to 3.

If insert

12. Press (+) /(–) to move to position to insert after.
13. Press 0 to 9 in keypad to enter patch number to insert, (has to be a two digit number).
14. Press STORE, STORE if you want to save the edit.
15. Press EDIT to return to 3.

If delete

16. Press (+)/(–) to move to position to delete.
17. Press '0' in keypad to delete patch stored at that position.
18. Press ALL/DEFAULT to clear sequence.
19. Press STORE, STORE to save edit.
20. Press EDIT to return to 3.

ROUTE AND ROUTE EDIT

Route is the second set of parameters that go to make up an MTM patch.

patch = process + ROUTE + effect.

ROUTE controls the connection of inputs to outputs inside MTM, both for triggers and MIDI data.

For example MTM is hooked up to an SDS 7 drum synthesizer. The pads are connected to MTM channels 1 to eight. The 'normal' routing of those pad trigger signals would be to the SDS 7's inputs channels 1 to eight. Channel 1 to 1, 2 to 2, 3 to 3 etc. This would mean that the pad connected to channel 1 would be playing a bass drum voice, the pad connected to channel 2 would be the snare etc. This 'straight through' route could be ROUTE 1.

You could then create a route which connects MTM's channel 1 input to channel 2 output, and channel 2's input to channel 1's output, and call this route ROUTE 2.

Then simply by swapping between route 1 and 2 the bass and snare pads would swap voices. Opening up a new avenue of experimentation for the drummer.

Route is also where you set up which channel MIDI data can be received and sent, and which note is received and sent. so each route can have a different set of eight MIDI notes associated with it, and thus as you step through MTM's routes, external MIDI voicing will be playing different sets of MIDI notes (chords.)

There are three parameters that you can program for each channel of MTM for each route. They are the trigger output channel number, the MIDI channel number, and the MIDI note number.

Editing Route

If you have learned how to edit patch, this is going to be simple, as the same structure of editing is consistant throughout MTM.

Enter edit route by pressing ROUTE EDIT from playback state..

As before at this point you can change the route to be edited if you wish, by pressing the (+)/(−) buttons.

displays:

route edit	01
01	

From this point you can press any one of the following buttons:

TRIG IN
MIDI CHAN/NOTE
TRIG OUT
MIDI NOTE

Each button will bring up the same display on the lcd, but with the particular parameter underlined by the cursor, ready to be updated.

The display will look like this:

route 01		trgin:1
ch01	ntC4	toutA

This is telling you that in ROUTE number 1, trigger input channel 1 is routed to trigger out channel 1 (the 'A' signifies that ins to outs are connected 1 to 1, 2 to 2, 3 to 3 etc) and that MIDI is sent and received for channel 1 on MIDI channel 1, MIDI note C4.

To alter any of the parameters, position the cursor under the item to be changed by selecting it with the appropriate button. Then use (+)/(−) to change the value.

E.g. to route channel 2 trig input to channel 8 output, and to change MIDI channel to 12, and MIDI note to C5:

Press TRIG IN change value to 2 using (+)/(−) or keypad.
Press TRIG OUT change value to 8 using (+)/(−) or keypad.
Press MIDI CHAN/NOTE and change value to 12 using (+)/(−).
Press MIDI NOTE and change value to C5 by using (+)/(−).

The display should show:

route 01		trgin:2
ch12	ntc5	tout8

NOTE that pressing the '0' button displays the MIDI note number and not the note name.

If you want all the trigger channels in this route to be sent down one MIDI channel, you can select MIDI channel and then change this, for each trigger channel, OR, you can press the ALL/DEFAULT button. This makes the parameter that you are changing apply to all trigger channels. When all trigger channels have the same MIDI channel, and the MIDI channel is selected to be changed, the trigger number displays 'A' – to show they are all the same.

The display will look like:

route 01		trigin:A
ch01	ntG4	toutA

Which shows that all triggers send MIDI notes down MIDI channel 1. Note also that all trigger outs are 'through'.

Pressing ALL/DEFAULT while changing the trigger outs will set all triggers to 'through', ie. 1 goes to 1, 2 to 2, 3 to 3 etc.

At any time while changing MIDI chan or MIDI note, you can use the key pad to immediately select a different trigger in. The key pad can also be used to change the trigger out associated with the correct trigger in only after the TRIGOUT button has been pressed. ie. while you are changing the trigger out.

Storing Routes

You store routes in the normal way, ie. store, select number, store. Note that user Routes are numbered 21-99.

SUMMARY OF EDIT ROUTE

To edit a route (from a playback state)

1. Press Route (ie. enter play route state)
2. Select a route with +, –
3. Press EDIT.
4. Press either TRIGIN, TRIGOUT, MIDI CHAN/NOTE, MIDI NOTE.
5. Change the data with +, –
6. Use the key pad to select different trigger channels (1-8).
7. Use ALL/DEFAULT to make all triggers the same (for the parameter selected).
8. Maybe go back to (4).
9. Press STORE.
10. Select using +, – where to put this new route.
11. Press STORE to save, anything else to abort.
12. Press EDIT to exit route edit. (returns to 2).

ROUTE DATA TABLE

Route Number 1

TRIG INPUTS:	Trg 1	Trg 2	Trg 3	Trg 4	Trg 5	Trg 6	Trg 7	Trg 8
MIDI Chan	1	1	1	1	1	1	1	1
MIDI Note	C4	E4	G4	B4	C5	E5	G5	B5
Trig Output	1	2	3	4	5	6	7	8

EFFECTS

MTM is an interface between signals that you wish to use to trigger instruments that actually make the sound that you hear. MTM is therefore in an excellent position to add things to that triggering information that weren't there in the original signal.

For example MTM could send a 'new' trigger shortly after it has sent the legitimate trigger, producing a single echo, (the receiving voice cannot know that it received two triggers but that you only hit the drum once). If MTM were to continue to send delayed triggers, each one at a lower dynamic than the previous, then a repeat echo effect would be produced by the receiving instrument.

These 'echo' signals can be sent to the trigger output of MTM or to MIDI out. Of course with MIDI out you can control the actual note that the instrument is playing, as well as the type and dynamic of the sound, so the echoes produced by an instrument controlled by MTM's MIDI could all be at different pitches! or as you strike the drum MTM could change the MIDI note in accordance with a pre-set sequence of notes, or build a C major chord, the number of notes in the chord in proportion to how hard you hit the drum, in short facilities that the percussionist could only dream of till now.

MTM EFFECTS – a list

ECHO

Echo normal adjust threshold, speed, decay and number of echo's.

Echo up in 1's – each echo is a semi-tone higher. Adjust threshold, speed, decay and number of echoes.

Echo down in 1's – each echo is a semi-tone lower. Adjust threshold, speed, decay, and number of echoes.

SEQUENCE

Sequence, echo stepped – each echo is a different note. You program the sequence of notes. Adjust note numbers, threshold, speed, decay and number of echoes.

Sequence, trigger stepped – each trigger steps the sequence onto the next note. Adjust note numbers, threshold, and number of notes in sequence.

Sequence, time stepped – each new note in the sequence is stepped through automatically. When a trigger arrives, the nearest note in the sequence is played. If the sequence is a series of descending notes, then a conventional 'run generator' effect is produced. Adjust note numbers, threshold, tempo and number of notes in sequence.

Sequence, auto stepped – each new note in the sequence is stepped through automatically. Adjust sequence note numbers, length of note, threshold, tempo and number of notes in sequence.

EDITING INPUT EFFECTS

CHORDS

By programming a different MIDI note for each channel of MTM a chord can be played when two or more of those channels are triggered together. In EFFECT edit you are allowed to program a complete chord for each channel of MTM.

The complete chord can be played as the trigger arrives, or you can program MTM to play only a single note in the chord, but add more notes as the trigger dynamic increases. (The louder you strike the drum the more notes the chord has.)

MTM has most of the standard chords pre-programmed, all you have to do is to choose the root of the chord, which type of chord and how it is played by MTM.

These chords can be programmed into the sequences as described previously. So you can see it would be possible to program chord riffs on each Simmons' pad that triggers MTM.

A list of the chord effects:

Split chord dual note – A two note chord is programmed. The first note (or root) is played as the pad is struck quietly, at a pre-programmed dynamic level the second note is played instead of the first. So two separate notes are produced dependent upon how you strike the drum. E.g. a low 'E' note sounds as the pad is struck softly – accents played on the drum play an octave higher.

Split chord multi-note – Same as above except that you can program six notes and use five different thresholds for those notes.

Layered chord dual note – The same idea as the split chord dual note described above, except that the second note is layered on top of the first note, rather than replacing it.

Layered dual chord – At low playing dynamics the root of the chord is sounded. At a pre-programmed dynamic the entire chord is played. You choose the root and chord.

Layered chord multi note – At low playing dynamics just the root of the chord is sounded, as the playing dynamics increase more and more of the chord is sounded, till at maximum dynamic a six note chord is played. You program the root and chord for the notes.

Effects are the third part of the set of parameters that go to make up an MTM patch.

PATCH = PROCESS + ROUTE + **EFFECT**

Some of the effects apply to trigger outputs and MIDI outputs, such as echo, but other effects (such as chord layering and sequencing notes) apply only to MIDI, as MIDI can control an instrument pitch, whereas a trigger pulse obviously cannot.

Those effects that apply to triggers and MIDI are selected by pressing the TRIG IN button, and are said to be 'input related'.

Those that are related to MIDI only are selected by the MIDI NOTE button, and are said to be 'output related'.

Enter process edit from play patch in exactly the same way as for process and route.

Press EFFECT then EDIT.

The display will show:

effect edit	01 01

You can change the effect to be edited at this stage if you require, using the (+)/(–) buttons as usual.

Press the TRIG IN button twice. If you are looking at effect number 1 then the display will look thus:

effec 01	trigin: 1
echo/seq:	off

This telling you that for effect number 01, trigger input channel 1, the echo and sequencer effects are switched off. Use the (+)/(–) buttons to see the status of the other channels. Alternatively use the keypad (1-8).

If you are still looking at effect number 1 then all the channels will be off.

To switch the input effects on press MIDI NOTE.

display:

effec 01	trigin: A
echo/seq:	off

The cursor will move to the second line, underneath 'off', and an 'A' appears instead of the channel number to indicate that ALL the channels are currently off.

To select an effect for channel 1 press the (+) button to cycle around the effects available. Each press of the (+) button will display a new effect.

The effects as described earlier are displayed:

effec 01	trigin: 1
echo normal	

effec 01	trigin: 1
echo up in 1's	

effec 01	trigin: 1
echo down in 1's	

effec 01	trigin: 1
seq echo stepped	

effec 01	trigin: 1
seq trig stepped	

effec 01	trigin: 1
seq time stepped	

effec 01	trigin: 1
seq auto stepped	

effec 01	trigin: A
echo/seq:	off

You can loop around these effects – forwards using the (+). When you have selected the desired effect for that channel, leave the selected effect displayed on the screen, and then press option. This then lets you adjust the parameters that go to make up the selected effect. (E.g. speed and number of echoes).

For demonstration select ECHO NORMAL.

display:

effec 01	trigin: 1
echo normal	

this is telling you that you are editing effect number 01, for channel 1, and that the effect you have selected for that channel is normal echo.

The parameters that you need to adjust to obtain an echo effect are: The **trigger threshold** for the echo (how big does the trigger have to be to start the echo?) The **echo time** (how fast is the echo?) **Echo decay** (how long does it take for the echo to die away?) **Echo number** (maximum number of echoes.)

To actually change these parameters press OPTION. Each press of OPTION will bring up a new parameter for 'echo normal'.

The parameters you can alter are: **Start threshold. Echo time, Echo decay,** and Echo Number.

displayed:

effec01 start thresh	trgin:1 :100
effec 01 time:	trgin:1 112m secs
effec 01 decays to :	trigin:1 80%
effec 01 number	trgin:1 :005

You can press (+)/(−) to change any of the parameters.

The above displays are telling you that for effect 1, channel 1 a normal echo is set up that wil echo any playing dynamics above the threshold of 100, that the echoes will be 112m secs. apart, and each new echo will be 80% as loud as the previous, and there will be a maximum of 5 echoes when hit hard.

Echo up in 1's

The same parameters are programmed in the same way for Echo up in 1's. During playback of the echo, each echo will be a semi-tone (or 1 MIDI note value) higher than the one before.

With echo normal on the display press (+) to display **echo up in 1's** and then press OPTION to see the parameters.

Start threshold
(Echo) time
(Echo) decay
(Echo) number

Echo down in 1's

Exactly the same set up and parameters as for normal echo and echo up in 1's, except each echo is sounded one semitone lower than the previous.

Select echo down in 1's with (+)/(−) and then press option to select parameter. Change the value of the parameter with (+)/(−).

Start threshold
(Echo) time
(Echo) decay
(Echo) number

THE FOLLOWING EFFECTS USE A SEQUENCE OF NOTES. THESE NOTES ARE PROGRAMMED DURING EDIT EFFECT SEQUENCE.

Sequence, echo stepped

The parameters for 'sequence, echo stepped' are as described for normal echo except that each echo is a new note that has been programmed in this effects' sequence.

Sequence Trigger Stepped

It is only necessary to program threshold and the number of notes in the sequence. You will be allowed to change values for echo, speed and decay, but they have no effect for this mode.

Sequence Time Stepped

Again, these parameters are programmed as described above for echoes, except that parameters for time and decay are not used.

Sequence Auto Stepped

As above except values for time are ignored.

ENTERING THE NOTES FOR EFFECT SEQUENCES

While in edit effect press sequence button, the display will change.

eff 01	sequence
tempo	:120

This is telling you that the sequence for effect 1 has a tempo of 120 beats per minutes. Note the cursor is under the value, use the +/– buttons to alter if required.

To program a sequence of different MIDI pitches, two pieces of information are required. – The notes pitch, and note length.

The note pitch is derived by entering an offset (nt-of) from the MIDI note number stored for the current route. This means that any sequence of notes can be applied to different pitches that have been programmed for the different MTM channels during route. IE. The same sequence is played back at different pitches, these different pitches are stored in route.

Route 1 CH1 Sequence (Offset values)

	MIDI note	Pos 1	Pos 2	Pos 3	Pos 4
	C5	+02	+07	+12	–02
Plays	C5	D5	G5	C6	A4

Route 2 CH1 Same sequence

	MIDI note	Pos 1	Pos 2	Pos 3	Pos 4
	G4	+02	+07	+012	–02
Plays	G4	A4	D5	G5	E4

The note length (len) is programmed for each position and is a choice of ¼, ½, ¾, 1/1 note values.

Press option the display will change.

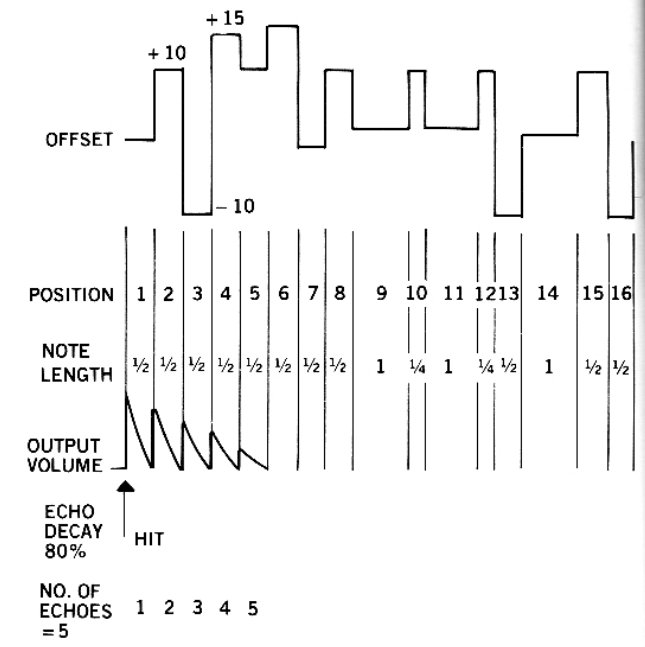
eff 01	seq pos :01
ntof: +00	Len :¾

The cursor is under sequence position 01, this is the first position for the sequence, use the +/– button to move forwards or backwards through the effect sequence.

To change a note offset, press option (note you will not be allowed to change position 1's offset because this is the root note programmed during route). Use +/– to change the offset, press option then +/– to change note length.

The effect sequence has a maximum of 16 positions, but the number of notes that are actually heard is generated by the number and decay programmed for the input effect. EG. You have programmed a sixteen step effect sequence but you have programmed the trigger-in decay to be so quick that only the first few notes are sounded. Or you could have restricted the sequence length by the number of echoes, if the number of echoes programmed is greater than 16 then the sequence will start again IE. If 32 was programmed, the 16 note sequence would be played twice.

EG. A 16 NOTE SEQUENCE



EFFECTS – EDITING CHORDS – OUTPUT EFFECTS

As described earlier there are **input effects** (those that apply to triggers as well as MIDI, such as echo), and **output effects** (those effects that apply to MIDI only such as chords). This section deals with the output effects, and they are:

Split chord dual note.
Split chord, multi-note.
Layered chord, dual note.
Layered dual chord.
Layered chord multi note.

Enter edit effect by pressing EFFECT, then EDIT.
Select the effect number that you wish to edit using the (+)/(–) buttons.Leave the display selecting effect number 1.

To select output effects press MIDI NOTE on the output section of the front panel. The output will display:

effec01	trginA
outef:	off

The cursor is underneath "off" and an 'A' appears in the top line indicating that currently all the channels are off.

To select one of the effects, press the (+) button.
The first press will change the display:

effec01	trgin:1
outef:	split dual

Subsequent presses will display:

outef:split mult

outef:layer dual

outef:layer chrd

outef:layer mult

And then a further press of (+) will take you back to 'off'. (You can use the (–) button to go backwards through the effects if you require.)

Editing split dual

Split dual means that you are programming a second MIDI note. The first or 'root' note is played when small trigger signals are received by MTM, (the pitch or MIDI note number is assigned during ROUTE). The second note replaces the first note at a pre-set dynamic level. So you have to program the second MIDI note number.

To program the second note, with the display as below:

effec01	trgin:1
outef:	split dual

Press OPTION, the display will change:

effec 01	trgin:A
note:	C0 –

Which means that the second MIDI note that will be played in this particular chord will be C0, – a very low note.

Use the (+)/(–) buttons to alter this note if required.

Change it to C5 – If you are hitting trigger 1, you will hear this note when you hit it hard.

You can press all at any time to make the displayed parameter currently being displayed, applies to all channels. Also pressing '0' will show you the MIDI note number instead of the chromatic note name.

Editing layered dual note

Use OPTION to return to the display 'split dual'. Press (+) to display layer dual. Editing is exactly the same as above – press OPTION, and then (+)/(–) to change the MIDI note value.

Don't forget to store edit by pressing STORE. STORE.

Editing split chord, multi-note
This time instead of programming a second note that is played when the drum is struck harder, you choose a chord, the notes of which are played separately as the drum is progressively struck harder. For example a major chord is chosen, and its root note is MIDI number 60 (middle C – programmed during route). As the drum is struck harder the notes of the chord will start to sound, each new note replacing the previous one.
C – E – G – C(oct) – E(oct) – G(oct).

Use the OPTION and (+) buttons to obtain the following displays:

effec 01	trgin: 1
outef: layer	mult

Then press OPTION to see the current chord chosen, press (+)/(–) to see the other chords that are available. They are:

chd:U:major	Upward major chord.
chd:D:major	Downward major chord.
chd:U:minor	Upward minor chord.
chd:D:minor	Downward minor chord.
chd:U:7th	Upward 7th.
chd:D:7th	Downward 7th.
chd:U:major 7th	Upward major 7th.
chd:D:major 7th	Downward major 7th.
chd:U:minor 7th	Upward minor 7th.
chd:D:minor 7th	Downward minor 7th.
chd:ascending 4ths	
chd:descending 4ths	
chd:ascending 5ths	
chd:descending 5ths	
chd:U:suspended	
chd:D:suspended	
chd:U:root + 5th	
chd:D:root + 5th	
chd:U:root + 4th	
chd:D:root + 4th	
chd:U:major 6th	
chd:D:major 6th	
chd:U:minor 6th	
chd:D:minor 6th	

Use '0' whilst displaying the chord to toggle between up (U) and down (D).

Layered, dual chord
Again choose the type of chord you require for this effect as described above.

The root of the chord (programmed in ROUTE) is played for small dynamics. The programmed chord comes in all together at a pre-determined dynamic level – Great for accents and chord stabs.

Layered chord, multi-note
This is programmed exactly as for split chord above, except that the notes of the chosen chord, instead of being played one after the other, are layered one on top of another, so at the highest playing dynamic all the notes of the chord are played together.

SUMMARY OF EDIT EFFECTS

Summary of edit effects

1. Press **effect** to enter play effect.
2. Press **edit** to enter edit effect.
3. Use + / – to select effect number.
4. Press **trig in, MIDI note,** or **sequence.**
5. Alter appropriate parameters.
6. Press **edit** to abort, or **store,** number, **store** to save.

Summary of edit effect – trig in

1. Press + / – to select echo/seq type.
2. Press **option** to select start threshold, time, decay and number parameters.
3. Press + / – to change data.
4. Press keypad 1 to 8 to choose new trigger channel.
5. Press **MIDI note, sequence, edit** or **store** as appropriate.

Summary of edit effect – effect sequence

1. Press + / – to change sequence tempo.
2. Press **option** to edit sequence position, note offset and note length.
3. Press + / – to change data over cursor.
4. Press **trigger in, MIDI note, option, edit, store** as appropriate.

Summary of output effects – MIDI note

1. Press + / – to change output effect type.
2. Press **option** to select note or chord.
3. Press + / – to change data.
4. Press **'0'** to see MIDI note number/chromatic name whilst displaying MIDI note.
5. Press **'0'** to choose up or down chord whilst displaying chord names.
6. Press **trig in, sequence, edit, store** as appropriate.

At any time press **All/Default** to make displayed parameter apply to all trigger channels.

PROCESS – A DESCRIPTION

PROCESS

Process is the last set of parameters that go to make up an MTM patch.

PATCH = **PROCESS** + ROUTE + EFFECT

It is the most complicated part of MTM as it has to translate all sorts of triggering signals that can be used to cleanly trigger all sorts of voicing systems.

Once you are into EDIT PROCESS you will be able to change the following parameters –

THINGS TO DO WITH INPUT TO MTM –

Threshold

input gain
% above previous trig
absolute threshold
% above another channel

Holdoff

holdoff time
dynamic holdoff

compress/expand

choose dynamic curve
minimum output

MIDI in

absolute threshold
choose dynamic curve
minimum output

THINGS TO DO WITH OUTPUT FROM MTM –

trigger outputs

pulse width
dynamic pulse width

MIDI out

fixed width
dynamic width

MTM's front panel is laid out so that those functions underlined above are a main function button on the front panel, the parameters listed below each heading are options (selected by the option button.)

Before altering process parameters here is a brief description of each parameter and how it is treated by MTM.

EXPLANATION OF THRESHOLD

Threshold is the level to which an incoming audio signal must reach before it is recognised as a trigger. This level is programmed for each channel in each process and allows you to exclude background noise etc.

This called absolute threshold. Other thresholds are % above previous threshold'' which only applies during the ''hold-off'' time and ''channel compare'' threshold – both of which will be described after the hold-off section.

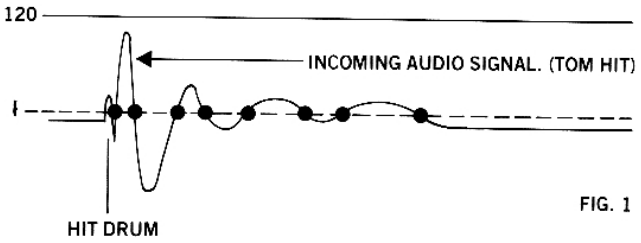


FIG. 1

Low numbers mean that quiet sounds will be recognised, higher numbers mean that a louder sound is required before it can be recognised.

HOLD OFF

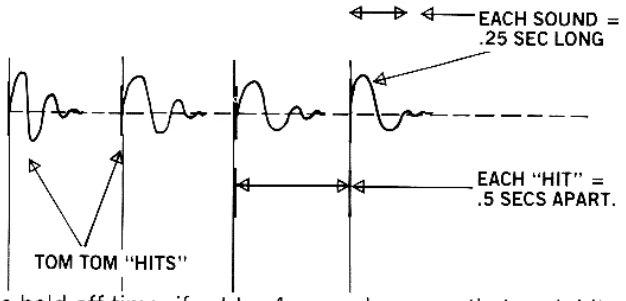
As can be seen from Fig. 1 any threshold that is set intersects a typical audio signal at more than one location.

In the case of the most sensitive threshold (value = 1 in the above example) the threshold is crossed in 8 places (in reality it would be many many more). It would be unacceptable to allow each of these points to be recognised as a trigger. Only the first trigger is required to set off (or trigger) the synthesizer otherwise the resultant sound would be ''graunchy'' or multiple triggered.

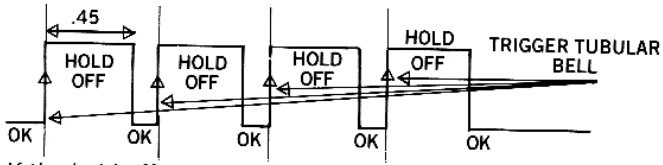
To gate out these multiple triggers you are allowed to program a ''hold-off'' period to suit the audio signal. During this hold-off no incoming audio signal will be accepted as a new trigger. Thus the hold-off can be defined as the time delay before a new trigger can be recognised. This hold-off can be programmed individually for each channel in each process and ranges from .008 seconds to approx 4 seconds.

This hold off type is defined in MTM as an ''absolute hold-off'' when % above previous threshold = absolute, as no new triggers are allowed during the hold-off period set (as already mentioned % above previous threshold and channel compare threshold may allow new triggers during the hold-off period.)

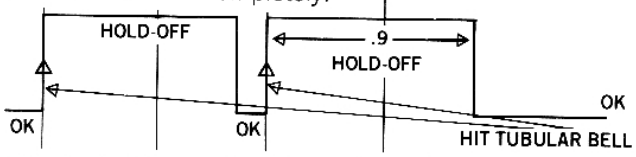
Example: A tom tom fill has been recorded on tape, each individual ''hit'' is half a second apart and each sound lasts for a quarter of a second. A tubular bell sound is to be triggered from these ''hits'' via midi.



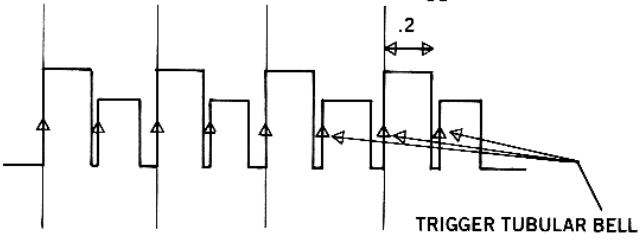
The hold-off time, if set to .4 seconds means that each hit will only produce one trigger (and hence one tubular bell sound).



If the hold-off was set to .9 seconds the second and fourth hits would be lost completely.



If the hold-off was set to .2 seconds, each tom tom hit would play the tubular bell twice, and the second ''hit'' would be less loud than the first because of the waveform produced by the tom tom (like an echo or double trigger).



%ABOVE PREVIOUS THRESHOLD

In the example above all new incoming audio signals are ignored during the hold-off period, and this is fine as long as the audio signals that you want to use as triggers are spaced apart in such a manner that they don't run into each other. Unfortunately this is very often not the case. Consider the following example –

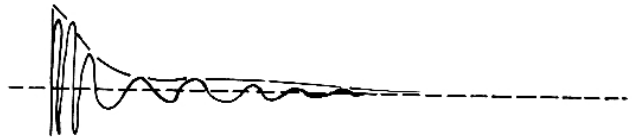
A good drummer can play a "closed roll" on an acoustic snare drum. Each individual strike of the drum is very close together. As close as 30 ms (milliseconds) or 33 strikes of the drum per second. These strikes could still be separated using absolute threshold as described, as long as the actual sound of the snare drum was shorter than 30 ms – which is a very short snare sound indeed!

Let's take an average snare sound of half a second (which consists of the snare being struck, the drum sounding, room ambience, reverb etc), if you were to set the hold-off time to .5 second to prevent multi triggering as the snare was buzzing and the reverb was twanging the sixteen hits of the drum that could be played in half a second during a closed roll would be ignored.

So, how can MTM tell the difference between the hit lasting .5 of a second and the sixteen or so hits that could occur in that time if a closed roll was played?

Luckily most audio sounds that would be required to "trigger" other sounds have a similar "shape". I.E. they have a definite "start" and this start portion of the sound is louder than the rest of the signal. All percussion sounds have this shape, the start portion of the sound being where the stick hits the drum or cowbell or whatever.

Here is a picture of the signal produced by a snare drum when it is struck –

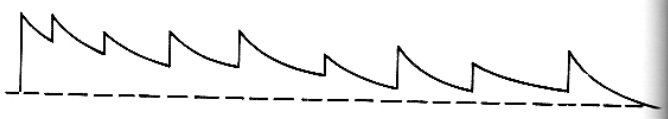


By joining the tops of the signal together the "envelope" of the signal is produced.

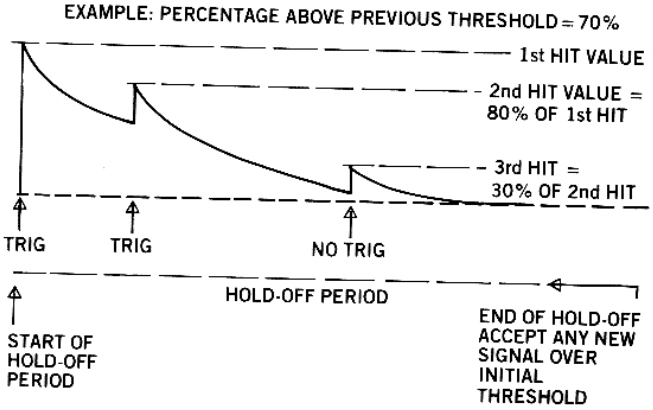
This envelope is the signal that MTM works on to determine when the snare drum has been hit. The first "spikey" part of the envelope is where the drum has been struck.

If the drum is struck again, two spikes would be produced, changing the smooth decay shape of the normal envelope.

The drum could be struck many times and each strike would produce a new spike in the decay curve of the previous hit.



You can program MTM to produce a new trigger when these spikes reach a certain percentage of the original trigger – this is called % above previous threshold.



By tailoring the hold-off and % above previous threshold to the type of audio signal that you wish to use as a trigger for synthesizers many different types of signals can be accommodated.

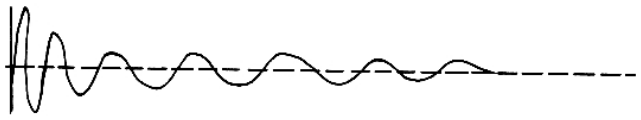
DYNAMIC HOLD-OFF

In the above examples it is assumed that each "envelope" produced by each hit of the drum is the same. This of course is not the case as the sound of the drum varies dependent upon how hard it has been struck.

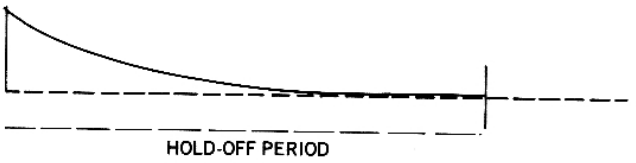
You will have noticed that the parameter of the sound that is important in deriving a clean, single trigger from each hit of the drum is the length of that sound, and this is the parameter that varies a lot when a drum is struck with differing amounts of force. I.E. Hit a tom tom hard and the sound takes a long time to die away, tap it lightly and a shorter, as well as quieter sound is produced.

Let's return to the example used in the previous section.

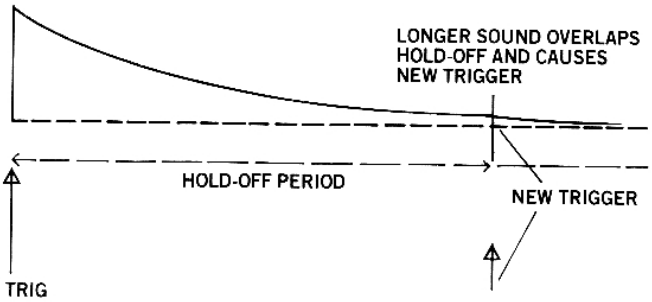
Here is a picture of the signal produced by a snare drum when it is struck –



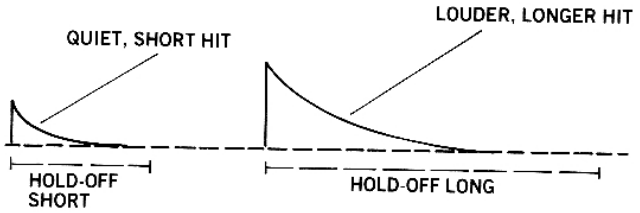
By joining the tops of the signal together the "envelopes" of the signal is produced.



If the drum is struck harder a longer envelope would be produced and hence the end of this signal will fall outside the hold-off period that has been programmed and MTM will think that the drum has been struck again.



This is where dynamic hold-off comes to the rescue. It enables you to program MTM to increase its hold-off time in proportion to the incoming signal level. Thus a loud sound will be assigned a longer hold-off than a short sound. This means that thresholds can be set up that are sensitive enough to pick up fast delicate hits on a drum, but not be swamped by a loud hit – thus preventing any false triggering that may have occured.



CHANNEL COMPARE THRESHOLD

There are occasions when signals from an adjacent channel may need to control whether a trigger has occurred on another channel.

For example – You wish to derive two triggers from a drum.
Trigger 1 = When you hit the drum head.
Trigger 2 = When you hit the drums rim.

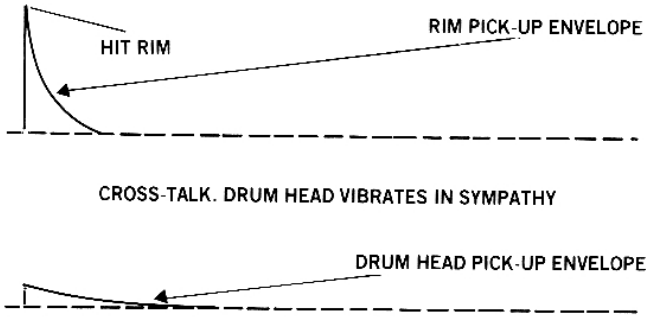
These two separate triggers can be converted by MTM to trigger two notes on a DX7 via MIDI.

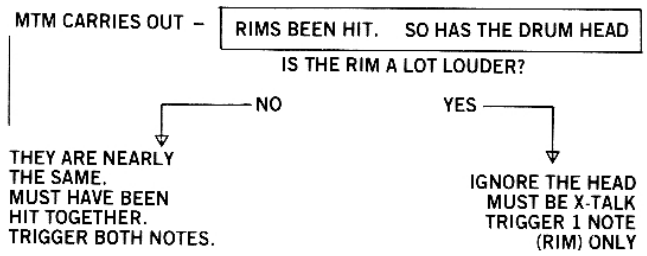
The problem is that the two signals are produced by the same drum and therefore cross-talk is inevitable. When the rim is struck the drum head vibrates in sympathy, thus fooling MTM into thinking it has been struck (but quieter). To a lesser extent when the drum head is struck the rim vibrates as well.

The solution is as follows – When MTM thinks the drum head has been struck (it has picked up a signal from the pick-up attached to the drum head) it looks at the rim pick-up to see if it is producing a larger signal. If it is, then it can fairly safely assume that the signal being produced by the vibrating drum head is as a result of cross-talk and can therefore be ignored.

How then can both notes be played simultaneously on the DX7 by hitting both head and rim (rim shot), when MTM will be ignoring the larger of the two signals?

The answer once again is programmability. You can program MTM to ignore signals coming from the drum head (in this example) when the rim is producing a louder signal, until both signal levels become very close – say within 10% of each other. This must mean that both rim and drum head must have been struck together as it is unlikely that there would be that much cross-talk in the drum.





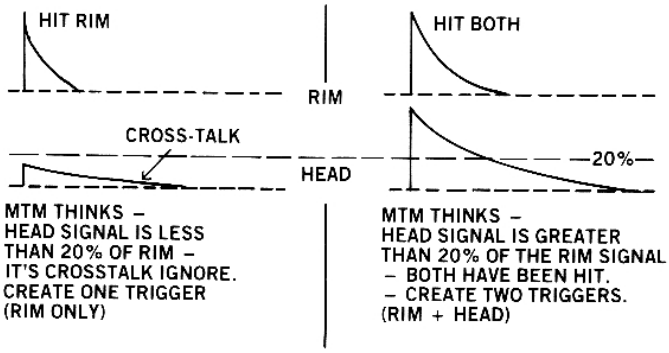
Of course you can programme the 'percentage-the-drum-head - has - to - be - of - the - rim - signal - before - it - is -recognised'-amount. (% above channel).

If it was 90% the drum would have to be hit 90% as hard as the rim before MTM thinks they have both been hit. If it was 100% both rim and head would have to be hit with exactly the same force for both to be recognised as being hit.

In reality programming this cross-channel threshold is simple –

1 Set the percentage to zero.
2 Strike the rim as hard as you are likely to hit it.
3 The drum head will obviously cause a trigger.
4 Keep raising the percentage whilst hitting the rim.

At some point the drum head will stop causing a trigger. This is the point where the maximum cross-talk the drum can produce is ignored by MTM. To make the drum head vibrate more than this, when the rim has been struck, you would have to have hit the drum head as well. And this would cause a trigger – which is what every body wants to happen!



SUMMARY OF INPUT SIGNAL PROCESSING

Gain – Sensitivity etc.

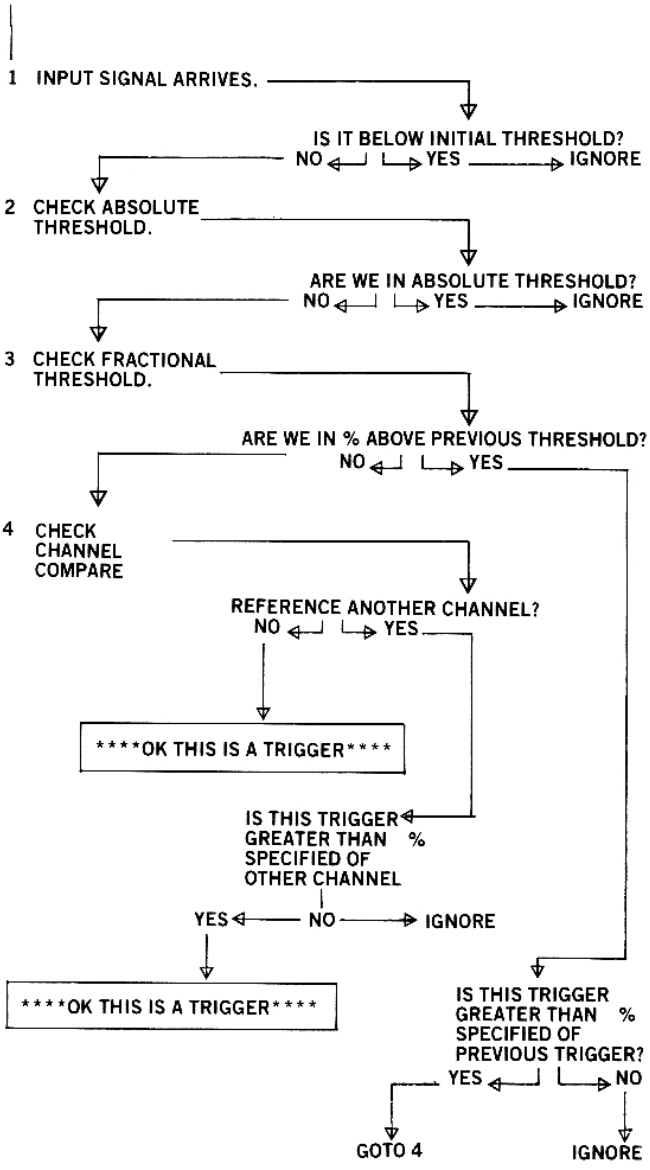
Before MTM receives a signal to be processed the signal passes though various pre-amplifiers.

These amplifiers are there to match the incoming signal so, for example a small signal from a dynamic microphone can be amplified to a level so as to cause a "maximum hit" dynamic from MTM, whilst the large signals created by the Piezo crystals in Simmons' jack pads can be attenuated so they don't overload MTM. (This cannot cause any harm, just the dynamic range would be 'squashed' into one band making all 'hits' sound the same, without any dynamic control. – The inputs are said to be 'saturated.')

Each individual input on MTM has a variable gain control on the front panel. These controls are not 'programmable', their positions are not remembered by MTM. They should be treated as pre-sets and be used to set the range of incoming signals so that the largest signal required causes the maximum trigger to be generated by MTM, but not so high as to overload.

Each input has two input gain ranges – low gain (line level), and high gain (mic level). This choice of range is under program control and can be individually set for each channel in each MTM 'process'.

The total process for MTM to recognise a signal as a trigger is as follows –



DYNAMIC CURVES

We have now come to the stage where MTM has recognised that a channel has 'received' a trigger.

To get to this stage the signal must have passed all the conditioning tests described in the preceding chapters (i.e. thresholds, hold-offs . . . etc).

MTM can ultimately do two things with this trigger.
1 Generate a dynamic trigger out signal (gate).
2 Generate a MIDI note 'on' signal.

The gate signal is a positive going pulse which can be used to trigger synthesizers via their 'gate in' or 'trig in' sockets.

The MIDI signal is a stream of data which can tell a MIDI equipped synthesizer which note to play.

Both of these signals are sensitive to dynamics. I.E. if MTM has received a large trigger then it will generate a large gate signal, and a loud MIDI note. If it receives a small trigger then a small trigger and quiet MIDI note will be generated.

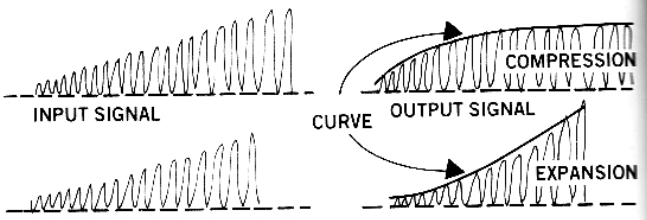
If the receiving synthesizer is sensitive to the varying gate and MIDI signals then the sound they produce will vary accordingly, and hence the player has dynamic control over these synthesizers by inputting dynamic signals to MTM.

COMPRESSION. EXPANSION. PROGRAMMABLE CURVES.

The above paragraph assumes that MTM, once it has recognised a trigger, passes it through without altering its' dynamic value. If the input signal is 'loudness value' 6 (on a scale from 0-10), then the output gate or MIDI signal will have that dynamic value as well.

You are probably aware of the terms 'compression' and 'expansion' as applied to audio waveforms – compression is where loud peaks are restricted (compressed) so as to reduce the total dynamic range of the sound whilst expansion is the opposite in that the peaks in the waveform are exaggerated to make them even louder.

Both compression and expansion can be represented as 'curves'. These curves represent the relationship between input signal and output signal.



The amount of compression or expansion can be represented by the shape of the curve. The point at which compression or expansion starts to affect the signal can also be represented by different shaped curves. These compression and expansion curves can be applied in exactly the same way to the trigger signals that have been derived by MTM from acoustic signals.

MTM has many pre-programmed curves that can be applied to the incoming 'triggers' that have been recognised by MTM.

Let's take our example of the snare drum again.

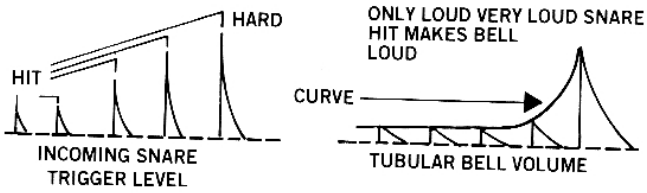
A signal from a snare drum is being used to trigger a tubular bell sound on a DX7 via MIDI.

If the incoming signals from the snare drum are not treated by dynamic curves, then as the drum is struck harder the tubular bell gets proportionately louder as well.

It may be desirable for the tubular bell sound to be very quiet for most hit levels on the drum, and only get loud when the snare is struck very hard.

MINIMUM OUTPUT DYNAMIC

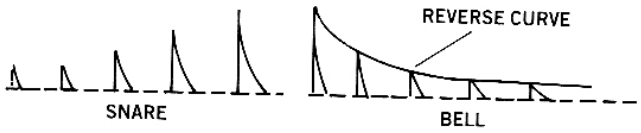
You would want to expand the incoming dynamics and adjust them by the following curve.



Different curves would have different effects on the tubular bell volume.

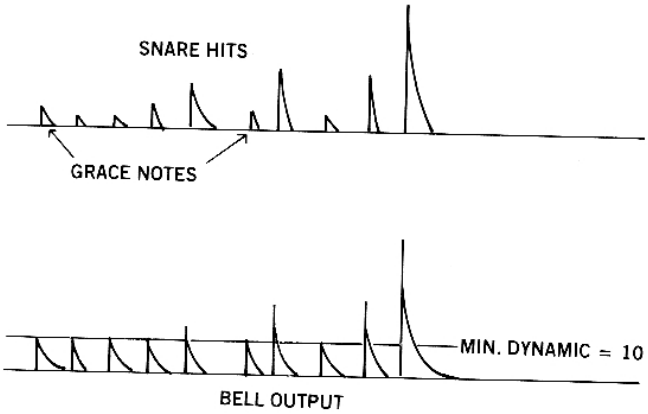
This one would actually reverse the dynamics totally, i.e. loud snare hits would produce quiet tubular bells, whilst quiet snare hits would produce loud tubular bell sounds.

You can set different dynamic curves for each channel in each MTM process



This is the minimum output level that MTM will play for any incoming signal level. It is used to make 'grace notes' sound that otherwise would be to quiet.

You can programme different minimum output dynamic values for each channel in each MTM process.



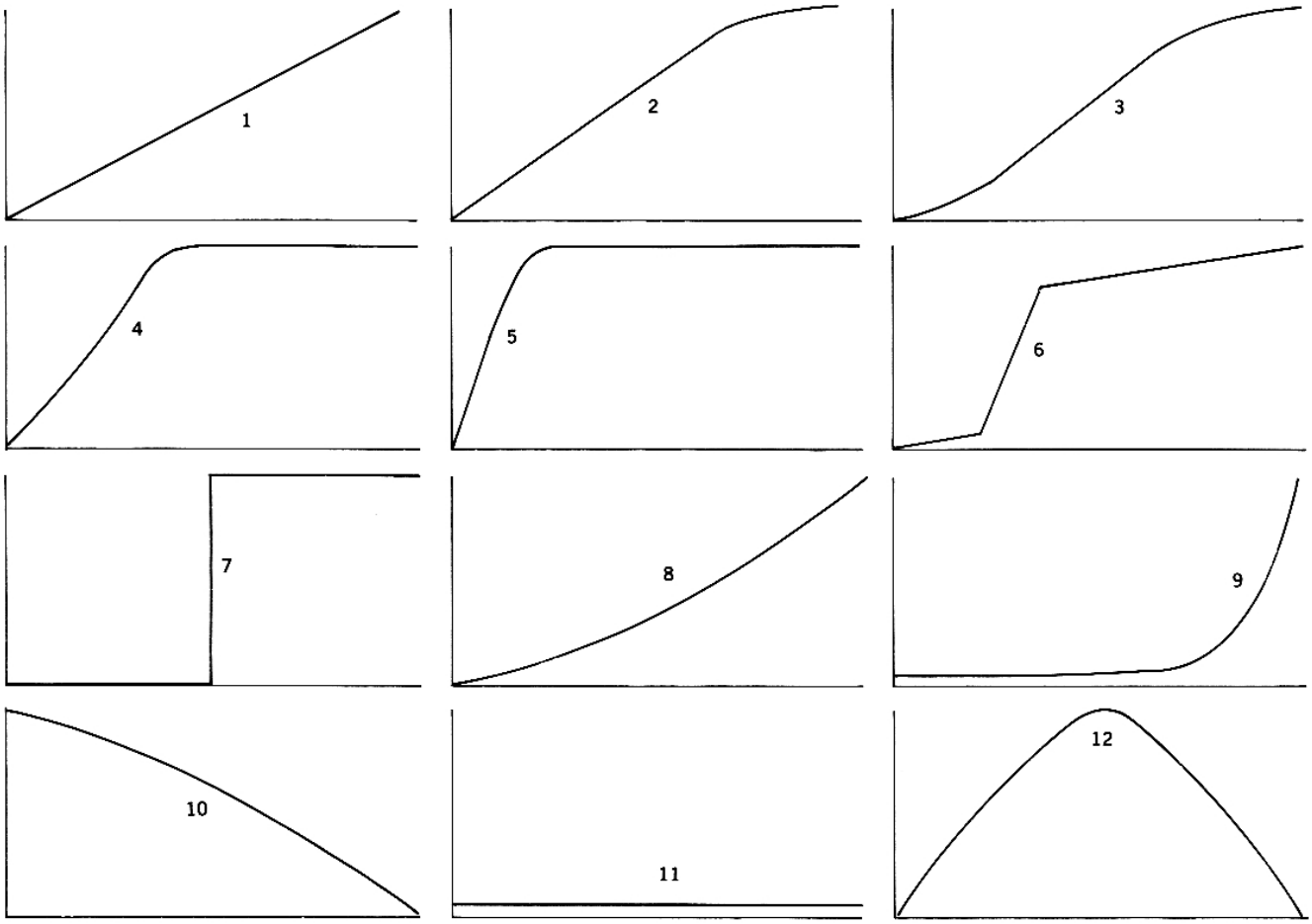
The values programmed for threshold, and minimum output dynamic, as well as the choice of compression/expansion curve. Apply to both trigger data and MIDI note data.

IE. Channel 1's trigger can have a different threshold, minimum output dynamic and curve to the channel MIDI note data.

MTM – DYNAMIC CURVES

Here is a list of dynamic curves that can be chosen to affect MTM's dynamic range. Some of them may seem a little odd, ie. The reversed curve or the up/down curve, but they can have spectacular effect in pre-recorded trigger signals when these are converted to MIDI notes.

1 Positive linear ramp.
2 Positive ramp with flat top.
3 S curve.
4 Sharp S curve – (gets loud easily).
5 Very sharp S curve (only very quiet dynamics are quiet).
6 Squared top and bottom ramp.
7 Dual level.
8 Exponential 1.
9 Exponential 2.
10 Negative ramp (small hits = loud).
11 Flat.
12 Positive/negative (medium hits loud, small and big hits quiet).



OUTPUT SIGNAL PROCESSING

There are two parameters associated with output processes. These are the fixed width and dynamic width of output signals (both trigger output for channel, and MIDI note data).

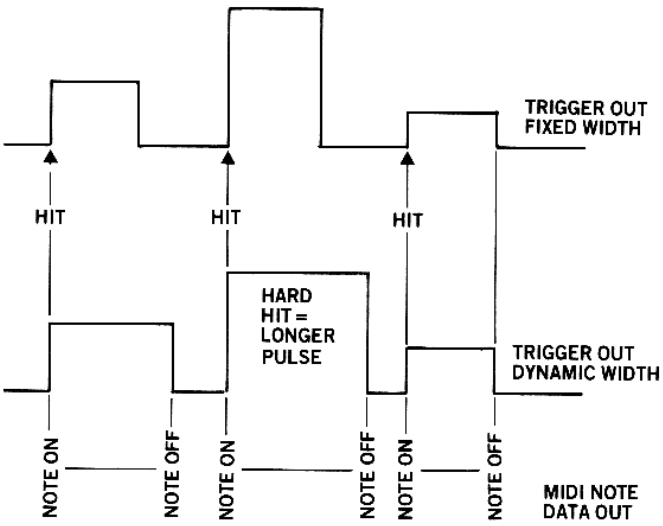
Fixed Width

The width of the output pulse can be thought of as the distance between the note starting (like pressing a note on a keyboard) and finishing (releasing that note). Indeed, old analog synthesizers used this 'gate on' 'gate off' signal to control its' envelopes, where as more modern synthesizers have replaced this signal with 'MIDI note on' and 'MIDI note off'. For electronic drums the 'gate on' or 'MIDI note on' signal is the only signal of interest because drums are struck and not pressed! – Although varying the width of the gate pulse can have a marked effect on the resultant sound.

IE. A short MIDI note will generally produce a short sound on the synth, whereas a long MIDI note will produce a long sound.

For drums it would be useful if this note on/off time could be varied with how hard the drum has been struck. This would mean that small hits would produce short sounds, whilst hard hits produce long sounds (very acoustic!).

The amount of change in width for changes in incoming trigger levels is adjusted by the dynamic width control.



EDIT PROCESS

Explanation of terms used during edit PROCESS

A detailed explanation of the terms and concepts behind MTM were given earlier but here they are again in short form.

Input gain
For each trigger input circuit on MTM you can program it to be high gain (to suit a microphone signal) or low gain (to suit line signal levels, or those coming off a Simmons' pad).

Absolute threshold
For each trigger input on MTM you can program a signal level below which all signals are ignored.

Percentage above previous threshold
Used to enable MTM to ignore an audio signal while it is 'dying away'. A new trigger will only be generated when MTM sees a signal that is 'x' percent (%) above previous trigger. – You program this percentage.

Percentage above another channel's signal (channel compare threshold)
You program channel 'a' only to trigger if its' signal is 'x' percent above channel 'b'. This is used to 'gate out' cross talk between two audio signals.

Hold-off time
You program a time period after each trigger. A new signal arriving during this 'hold-off' time may be ignored.

Dynamic hold-off time
The hold-off time gets longer automatically, in proportion to the input signal. I.E. louder sounds have a longer hold-off than quieter sounds. You program the difference between the longest and shortest hold-off time.

Dynamic curve
For each trigger input on MTM you can choose a 'dynamic curve' which will 'compress' or 'expand' the dynamic range of incoming trigger signals.

Minimum output
For each channel on MTM you can program a 'minimum trigger' level. I.E. the trigger output will always be this level or louder no matter how small the input signal level becomes.

Pulse width
For each trigger out on MTM you can program the pulse width of the trigger.

Dynamic pulse width
For each trigger output on MTM you can program the pulse width to vary with the incoming dynamic. The bigger the trigger, the longer the output pulse becomes.

MIDI in absolute threshold
The same as described for trigger inputs above, but the MIDI in signals.

MIDI dynamic curve
Chooses the same curves as above, but affects MIDI dynamics.

MIDI minimum output
Sets MIDI minimum level out, as for triggers above.

MIDI fixed width
Sets the time between MIDI 'note on' signals and MIDI 'note off' signal.

MIDI dynamic width
Varies the note on – note off time, in proportion to MIDI dynamic level. (The louder the sound, the longer the MIDI note).

PROCESS EDIT – THE BUTTON PUSHING

The front panel of MTM is laid out as far as possible, as a graphical representation of signal flow through the instrument.

During most editing in MTM you can make the decision to edit all eight channels at once, or change each channel individually.

For example, you may want to set the absolute threshold to be the same for all eight channels, because the input signals are similar. It would be tedious to have to change each channel separately, so you deal with them all together. (The letter 'A' will appear in the display when this happens).

ENTERING PROCESS EDIT
From the 'switch on' state press PROCESS and then EDIT. The display will change to:

process edit 01
01

The 01 in the display tells you that you are editing process number 1. At this point you can change the process number if you wish by pressing the (+) or (–) buttons. (If you want to abort the edit, press EDIT again and you will be returned to PLAY PROCESS.)

At this point you can choose which parameter you wish to change. Referring to page 34 you can see that we can change things to do with INPUT – **(Threshold, Hold-off, comp/exp, MIDI in,)** and OUTPUTS – **(Trigger and MIDI output widths).**

First of all we will look at how you program MTM's INPUTS.

Threshold is first, and it has four parameters associated with it – input gain, absolute threshold, % above previous trig, and % above another channel.

Threshold – input gains
With 'process edit 01' being displayed, press TRIG IN. (Located at the left hand side of the graphics panel).

The display will change:

proc 01 trigin:1
input gain : line

The display is now telling you that you are editing PROCESS 1, and that the INPUT GAIN for TRIGGER IN CHANNEL 1, is set at LINE level. (I.E. it is low gain suitable for high, line level signals or Simmons' jack pads.

Note that the cursor is under the number **1.** This indicates where changes will occur on the screen.

Look at the gains set for the other 7 channels by using the (+) or (–) buttons. You can also enter the channel number directly, by using the keypad.

As you cycle around the 8 channels you will notice that they are all set for LINE level.

To change a channel to MIC level press THRESHOLD (on the left side of the panel).

You will now see that the cursor has moved under the 'e' of LINE and the letter 'A' has appeared to replace the channel number at the top the screen.

The cursor indicates that this is the information that you can change, and the 'A' tells you that this currently applies to ALL each channels of MTM. (They are all set to line level.)

To change the channel to MIC level press (+) or (–). (As there is only a choice of two levels – mic or line – subsequent presses 'toggles' the input gain from MIC to LINE.)

proc 01 trigin:1
input gain : mic

As soon as you change one of the channels to MIC level, the 'A' disappears and is replaced with the channel you are programming (because 1 channel is now at mic level, the other 7 are still at line level, therefore they are not All the same.

Select a new channel by pressing the keypad (1-8), you now press (−)/(+) to change the gain of the channel selected.

Go through all the channels like this until they are all set at MIC level. (Did 'A' appear when you changed the final channel?).

There is a faster way to change all eight channel gains at once, press THRESHOLD so the display looks:

```
proc 01                          trigin: A
input gain              :            mic
```

This is telling you all input gains are at mic level.

Press (+).

You have now changed one of the channels to LINE level. ('A' has been replaced with the channel you are programming – it does not matter which one).

Now press ALL/DEFAULT. The display will now show:

```
proc 01                          trigin: A
input gain              :           line
```

All eight channels are set to line level.

Summary of button pushing to change input gains.

From switch on, (play patch).

Press PROCESS. EDIT. TRIG IN – (+)/(−) to select channel.

Press THRESHOLD. (+)/(−) to select mic or line.

Press ALL to make all channels either mic or line.

Press keypad to select other channels.

Press EDIT at any time to return to start (play process.) Any changes that you made to the process so far are lost. To save your changes you have to STORE them.

Store the edited process by pressing STORE, the display will change to STORE PROCESS NN, you can now change the process number if you require.

To finally store the change press STORE again. The display will momentarily change to STORING PROCESS NN. before changing back to process edit.

Selecting the other options in THRESHOLD.

Input gain is only one option available to edit from **threshold,** the other options are – absolute threshold, % above previous trigger and % above other channel.

To select any one of these options enter process edit and select trigger in, press threshold.

displayed:

```
proc 01                          trigin: 1
input gain              :           line
```

Now press OPTION/SELECT.

displayed:

```
proc 01                          trigin: 1
absol thresh:                       001
```

Press OPTION again.

displayed:

```
proc 01                          trigin: 1
% above prev            :            99
```

Press OPTION again.

displayed:

```
proc 01                          trigin: 1
% above                       ch:-   :no
```

Pressing option again will take you to edit input gain. So you can 'cycle' around all four **threshold** parameters for each channel, by pressing the OPTION button.

Press keypad (1-8) to select next channel, and then press OPTION, and look at the four parameters for the next channel selected, and again for another channel.

As you can see its very simple to see all the parameters associated with the input's threshold using the option button.

You can also cycle around all 8 channels, looking at a single option.

For example, to look at all the absolute threshold values for the eight channels, enter process edit, select trig in, press option to obtain absolute threshold, and then use the (+)/(−) buttons (or the keypad) to cycle round the channels 1 to 8.

Use the option button and (+) to look at the % above previous values.

EDITING
−ABSOLUTE THRESHOLD, % ABOVE PREVIOUS TRIGGER AND % ABOVE ANOTHER CHANNEL.

The editing of these three parameters is handled in a similar way to that described for INPUT GAINS on page 27.

Absolute threshold
Enter process edit, select trigger in, select absolute threshold with the option button. Select the channel you wish to edit, and then press threshold.

The cursor will drop down underneath the number to the right of 'absol thresh:'. Use the (+)/(−) buttons to alter the number if required. Select the next channel to edit by pressing trig in, or the keypad (1-8).

If you want to change all channels to a specific absolute threshold, adjust any channel to that value and press ALL. The channel number will disappear and the 'A' will appear, indicating that all channels have been changed to that value.

To store the edit press store, choose a new process number if required and then press store again to save the edit.

% above previous trigger.
Edit this exactly the same as the absolute threshold, described above, except that the numbers have a different range, and that after 99 the letters 'abs' (for absolute) will appear. When abs(olute) is programmed for any channel then no new trigger can appear during the hold-off period.

% above another channel
The editing of this parameter is once again the same, except this time you have to choose a channel for reference. Enter edit % above channel, use (+)/(−) to choose the channel you are comparing signals to (obviously you cannot reference the channel you are programming) and then use option to move the cursor to the percentage figure. Alter the figure using (−)/(+).

15 represents maximum cross-talk rejection. 1 is minimum cross-talk rejection. OFF (ie −) is no channel compare.

Store any updates by pressing STORE (change process no.) STORE.

EDITING HOLD-OFF − hold-off time, dynamic hold-off
The editing of HOLD-OFF is carried out in exactly the same way as for Threshold.

Enter edit process, press trigger in, and then select hold-off by pressing HOLD-OFF TIME.

displayed:

proc 01	trigin:A
h.off	0040m secs

Which is telling you that all the inputs have a hold-off time of 40 milliseconds.

To alter the hold-off time use the (+)/(−) buttons as before.

To alter the channel press trig in and use the (+)/(−) or use keypad (1-8).

Press OPTION to program the DYNAMIC HOLD-OFF time.

display:

proc 01		trigin:03
dynam h.off	:	30

This is done in exactly the same way using the trigger in, (+)/(−) buttons or keypad (1-8) to change channel, and hold-off time, (+)/(−) to change the value.

EDIT COMPRESS/EXPAND − dynamic curve, minimum output
Enter edit process, press trigger in, press COMPRESS/EXPAND.

display:

proc01	trigin:A
curve	:01

This is telling you that you are editing process number 1, and that dynamic curve number 1 applies to All the inputs.

As before use the (+)/(−) buttons to select a new curve. Press trig in and then (+)/(−) or keypad to change the channel. The All button can be used to make a curve apply to all eight channels.

Press OPTION to select minimum output. Again use trig in and the (+)/(−) buttons or keypad to select the channel, and compress/expand, (+)/(−) buttons to change the minimum output.

display:

proc 01	trigin:A
min output	:021

EDITING MIDI IN – absolute threshold, dynamic curve, minimum output.

As with the audio trigger signals, you can program absolute threshold, dynamic curve and minimum output for the midi data coming in to MTM.

These three parameters apply to ALL the MIDI data. An A will appear in the display all the time indicating that this is the case. The actual MIDI channels and notes that the processing applies to are programmed during ROUTE.

You should be familiar with the structure of editing MTM by now.

Changing these three parameters is carried out in exactly the same manner as previously described, except you press the MIDI CHAN/NOTE button instead of the trigger in button.

Briefly, to edit **absolute threshold.**
Enter process edit. Press MIDI CHAN/NOTE.

proc 01	MINI in:A
absol thresh	:100

Use (+)/(−) to change absolute threshold for the MIDI data for process number 01.

To choose a **dynamic curve** for MIDI:
(Enter process edit, press MIDI channel/note) press COMPRESS/EXPAND. Use the (+)/(−) buttons to choose a new curve.

proc 01	MIDI in:A
curve	:03

This means that MIDI dynamics have been compressed by curve 3 (for process number 01).

Press OPTION to select MINIMUM OUTPUT level for MIDI in.

proc 01	MIDI:A
min output	:009

Use the (+)/(−) buttons to change the minimum output level for MIDI.

EDITING THINGS TO DO WITH OUTPUT FROM MTM

Referring back to page 36, there are four parameters that can be altered which affects MTM's output in PROCESS. Two for trigger outputs and two for MIDI out.

They are essentially the same parameters for both trigger and MIDI. I.E. the pulse width for the trigger outputs and MIDI (the distance between note on and note off), and the dynamic pulse width, i.e. the amount of variation in the pulse width due to variation in dynamics. The higher the dynamic, the wider the pulse, or the longer the delay between MIDI note on and note off.

All four parameters are programmed in the familiar way:

Enter process edit (if you're not already there).

Press the TRIG OUT button. Choose whether you want to edit PULSE WIDTH or DYNAMIC PULSE WIDTH using the OPTION button. Press the WIDTH button and (+)/(−) to alter the value, and the TRIG OUT button and (+)/(−) to change the channel, or use the keypad (1-8).

displays:

proc 01 p.width	trigout:7 0040m secs
proc 01 dynam width	trigout:7 :020

This is telling you that process number 1, channel 7 trigger output has a static pulse width of 40 milliseconds, increasing by 20% at the highest dynamic.

Editing MIDI OUT fixed and dynamic width.

The process is exactly the same as described above for the trigger outputs, except that it applies to MIDI out data (programming the delay between sending note on and note off information and therefore controlling the length of the resulting sound.)

From process edit, press MIDI NOTE. Use the OPTION button to choose FIXED WIDTH or DYNAMIC WIDTH and use the (+)/(−) buttons to alter the values. (Note the appearance of 'A' in the display, indicating that this applies to all the MIDI data for this process number.)

displays

proc 01 f.width	MIDI out:A 1000m secs
proc 01 dynam width	MIDI out:A :80

DON'T FORGET TO STORE THE PROCESS IF YOU WANT TO SAVE YOUR EDIT.

THESE PARAMETERS OF PROCESS CAN BE REPRESENTED BY THIS CHART FOR EACH PROCESS NUMBER: Each blank square would contain a number that represents the value set for that particular parameter.

INPUTS

		TRIG 1	TRIG 2	TRIG 3	TRIG 4	TRIG 5	TRIG 6	TRIG 7	TRIG 8
THRESHOLD	INPUT GAIN								
	ABSOLUTE THRESHOLD								
	% ABOVE PREVIOUS								
	% ABOVE CH AMOUNT								
HOLD-OFF	HOLD-OFF TIME								
	DYNAMIC HOLD-OFF								
COMPRESS EXPAND	CURVE								
	MINIMUM OUTPUT								

OUTPUTS

OUTPUTS	PULSE WIDTH								
	DYNAMIC PULSE WIDTH								

MIDI IN

ABSOLUTE THRESHOLD	
CURVE	
MINIMUM OUTPUT	

MIDI OUT

FIXED WIDTH	
DYNAMIC WIDTH	

WHAT IS MIDI! – INTERFACE TO OTHER INSTRUMENTS

MIDI stands for Musical Instrument Digital Interface and is a standard interface that allows many different types of instruments from several different manufacturers to be connected together. These instruments include keyboard synthesizers, drum machines, recorders/sequencers, effects units and electronic drum kits.

Information is transmitted and received between these instruments via 5 pin DIN connectors. This information is in the form of a 'serial stream', in other words all the information is sent one after the other in a serial form. This means that only two wires are needed to send and receive MIDI, although a 5 way connector (the din standard plug) is used for MIDI.

In the same way that a radio can be tuned into many stations (although the signals are being received down the same aerial), different instruments can 'talk' (transmit) or 'listen' (receive) on different MIDI 'channels'. There are 16 MIDI channels.

This enables many instruments to be physically linked together, and then 'switched' in and out, by changing MIDI channels.

An example – the SDS 9 electronic drumkit is connected to two MIDI voices, an analog synthesizer, and a digital synth. As the drum kit is played signals are sent down MIDI which tell the synths to play.

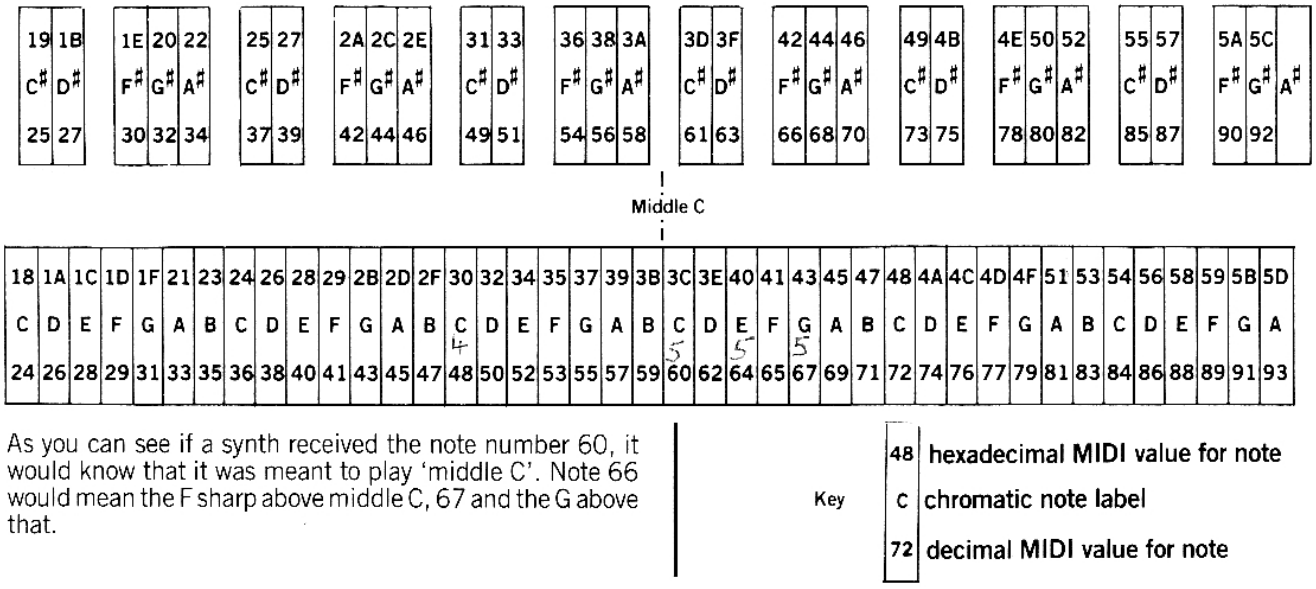
The SDS 9 has six 'drums' bass, snare, rim, and three tom toms. You want to add the sound of the analog synth to the bass and snare drums, but the digital synth to the rim and toms.

To separate the two sounds, you could program the SDS 9 to 'transmit' the bass and snare drums on MIDI channel 1, and the rim and toms on MIDI channel 2. Then if you programmed the analog synth to 'receive' on MIDI channel 1 and the digital synth to 'receive' on channel 2 the sounds would be separated as desired.

MIDI does this by sending the following messages down MIDI – Channel 1 data hit bass . hit snare, hit bass . . . Channel 2 data hit low tom . hit rim . hit hi tom . . . etc. And of course the analog synth is only listening to channel 1, and the digital synth is only listening to channel 2.

In reality the receiving synths have no idea that a 'bass drum' or a 'tom tom' is playing them. All they receive is a stream of numbers that tell them what note to play, when, and how loud to play it.

Here is the chart that converts MIDI note information to the standard chromatic scale:



As you can see if a synth received the note number 60, it would know that it was meant to play 'middle C'. Note 66 would mean the F sharp above middle C, 67 and the G above that.

MIDI TERMINOLOGY

The MIDI note numbers which set the note to be played are also accompanied by numbers which tell the synthesizer when the note starts (note on) and when it stops (note off), as well as how loud it is (dynamic).

You can also send a signal down MIDI that will tell the 'listening' synthesizer to change program, i.e. to change its' sound to a new pre-programmed sound.

You won't be surprised to learn that this signal is also a number, and that number is the program that the slave synthesizer will change to. If you send 'program change 34', then the slave synthesizer will change to program 34, whatever that has been programmed to be.

Channel, note, and program information are the three areas that MTM manipulates. Each MTM channel can send or receive a specific MIDI note, down a specific MIDI channel, at varying dynamic levels, and note lengths (the distance between note on, and note off).

Each MTM patch can have a different set of MIDI notes and channels, and can send a different program change number. Each MTM channel can send MIDI notes to play various chords, echoes, sequences, and other effects on the slave synthesizer.

You will come across other aspects of MIDI, sooner or later, here is a brief description of them:

NAMES USED – TERMINOLOGY FOR MIDI

When using MIDI equipped equipment you will find there are lots of buzz words and jargon used to describe things. This section is to explain some of this MIDI terminology.

Note information: when a MTM is triggered, information is sent down MIDI saying that a channel has come on (note on) and when a voice has been released (note off). This note information is sent down a Channel, it specifies which note it is and how hard it has been triggered.

Channel: there are 16 MIDI channels that can be used for transmitting MIDI information. The data all goes down the one MIDI cable, but can be directed to, and responded to, by assigning (selecting) channel numbers.

Basic channel: one channel is always assigned by each instrument to be its' basic channel. This channel is used for the information that affects all of its' voices. MTM does not have a basic channel.

Omni, Poly, Mono: these are used to describe how an instrument's voices respond to MIDI information. They basically describe whether the instrument ignores channel information, whether it responds to just one channel, or whether each voice is assigned a different channel.

Note on: describes an event that represents a voice starting to sound. Part of note on information is how 'hard' the voice is to be sounded, and part is which note (the note number) it should play.

Note off: describes an event that represents that a voice should now stop being sounded. Part of the note off info is how 'hard' the voice should stop being sounded, and part is which note (the note number) should now be released. (For example when a key is released from a keyboard, and how quickly it is released). Note off info is of limited use to percussion synthesizers, and is normally ignored.

Note numbers: each key of a keyboard has been allocated a note number by the International MIDI Association. Middle C has been defined as 60 (decimal), the C an octave below being 48 (decimal), etc. For percussion synthesizers (e.g. the SDS 9) it is usual to assign one MIDI note number for each drum (or voice).

Program change: when equipment changes patch (or a kit for the SDS 7/9), program change information can be transmitted via MIDI, thus allowing several synths connected together to change patch simultaneously.

Controller change: when two synthesizers are connected together it is often required for the performance controls on one to control similar controls on the other. This control information is sent via MIDI as controller changes.

System exclusive: this is used to transmit and receive special information between instruments from the same manufacturer. Examples of system exclusive data is data dumps etc.
Each MIDI equipment manufacturer can have a special number allocated – their MIDI ID, which allows their equipment to recognise its' own data, and to ignore other data. SIMMONS MIDI ID is 18 (decimal).

System real time: there are several real time messages that allow synchronising of drum machines, sequencers etc. MTM does not use any real time messages.

DUMPING TO AND LOADING FROM CASSETTE

Once you have programmed your own patches into MTM it would be wise to make a cassette copy of the data. Or you can use the cassette dumping facility to build a library of patches to suit every occasion and every instrument.

Once you have a copy of MTM's data on cassette, you can load it back into MTM at any time using the cassette load option.

There are three options that you can use that are associated with the cassette. They are Cassette dump, Cassette load and Cassette verify.

From the 'switch on' state, press cassette. The display will change to:

cassette dump?

Press OPTION to see the other two options:

cassette load? . cassette verify?

Pressing OPTION again will bring you back to cassette dump? Answering YES (the + 1.YES. button) will start the operation.

Cassette dump

Connect MTM's 'CASS' socket to your cassette recorders 'mic input' and 'earphone output' using the din to mini jack lead supplied.

During cassette dump, MTM converts the data into a serial stream consisting of two tones, and sends this signal out to the cassette socket (on the back of the unit).

The cassette recorder records the data as normal. To add a long leader tone to the beginning of the dump press the ALL/DEFAULT button before MTM starts to dump. This 'leader' is useful when setting the cassettes playback level during loading. During cassette dump MTM displays the following message:

dumping	cass
doing patch	nn

As MTM dumps more data the display will change: doing process, route, effect, sequence, MIDI.

Cassette verify

After you have copied MTM's data you can check to see if everything has been recorded OK, by verifying the cassette copy. This checks the data on the tape against the data stored in MTM, and passes it if it is OK, or fails it if a mistake is found.

MTM displays 'verifying cass' during the verification process, and displays 'tape read error' it it detects a fault in the cassette, or 'mem verify error' if it finds a difference in MTM's memory. If all is OK then you are returned to play patch.

Cassette load

Re-wind the cassette tape to the beginning of the data, start the tape and press 'YES' to cassette load? When MTM picks up the leader tape it will display – –. Adjust the tape play back so that – – is steady. When MTM picks up the data o – o – will flash and MTM will display 'loading cassette'. The data loaded from cassette will overwrite the old data stored in MTM's memory.

Aborting cassette operations

At any time you can press 'CASS' to return to the playback state.

USER AREA INITIALISE

User area initialise –
The initialisation of the user area copies sections of the factory patches, processes, routes and effects into the user area.

The factory settings are described elsewhere in this manual.

For the SDS 7 initialisation the first 16 patches are designed to match the SDS 7 XLR pads kit set up. (See SDS 7 – MTM – MIDI hook up).

User area initialisation
MTM comes supplied from the factory with a set of data loaded into the user area so that you do not have to start from scratch when you are programming up MTM patches.

As you continue to use MTM you will of course start modifying the user area to suit your own needs. You can of course dump to cassette tape the user area at any time to store your own personalised edits. You can however, return MTM to its initialised state, ie. with all the data that it was delivered to you as programmed in the factory.

MTM has two initialising modes. One initialises the user memory to cope with the most useful patches for an SDS 7 set up, the second initialisation sets MTM into a demonstration mode. You can initialise MTM at any time but remember any changes that you have made in the user area will be over-written with the initialised data.

To initialise MTM to either one of these states, start by holding down the default button whilst switching the power on, MTM will then respond with the following message:

Do you want to init SDS 7 data? At this point you can say 'yes' by pressing the 'Yes' button or press Option, in which case MTM will respond with the following message – Do you want to init demo data? Pressing Option again will bring up two similar messages except instead of initialising data, MTM will carry out a verification; that is it will check the user area against the data it has stored in its permanent memory.

Answering Yes to any of the questions will start the process of initialisation or verification. In the case of initialisation a message 'I will overwrite user data. sure?' will appear – it is a safety net, answering Yes to this question will actually overwrite the data.

Press any button at any time to abort the verification or initialisation process.

If everything goes okay, various messages will be displayed. In the case of initialisation the message will be 'Overwrite passed ok'. In the case of verification the message will be 'verify init data passed ok'. If something has gone wrong MTM will display the following message.
Failed! followed by a series of numbers which references an engineering to a particular location in memory where the failure has occurred. If this occurs refer to a qualified Simmons Service Centre.

SDS 7 HI HAT

The SDS 7 uses an OPTO-coupled Hi Hat pedal which produces a 'DC' signal proportional to the position of the pedal. (open, ½ open, closed, etc).

This is used by the SDS 7 to shorten and lengthen the Hi Hat sound as well as trigger the sound on closing the pedal. This means that a different type of signal has to be handled by MTM and passed down MIDI. This uses a signal called 'Controller change' rather than the 'note on, note off' signals used for normal triggering.

To switch CH8 into SDS 7 Hi Hat mode press MIDI, and then press Option until the display shows:

MIDI CH8 = TRIG

Press YES and the display will change to:

MIDI CH8 = HI HAT CHAN: 01 CONT: 001

Which means that CH8 is now configured for an SDS 7 Hi Hat pedal, and will be sending the position of the Hi Hat as a controller change.

MIDI supports 127 different 'controller numbers'. In the above example controller number 1 is being used and is being sent (and can be received) on MIDI channel 1.

To change the channel No. press +/–. To change the controller number press Option and then +/–.

To exit hit the patch button – your changes take effect immediately.

To switch CH8 back to a normal 'trigger channel' select MIDI channel 0 as described above. When you try to do this the display will show 'MIDI CH8 = TRIGH' once again. Exit by pressing patch.

NOTE: CHANGING CH8 TO HI HAT (CONTROLLER) OVERIDES ALL SET UPS FOR TRIGGER CH8 INFO SET UP IN ROUTE.

MTM MIDI GLOBALS

The following set ups affect all MTM patches.

IE. They are global.

They are:

1. For MTM receiving MIDI data.

a. Ignore channel information Yes/No.

b. Ignore MIDI effects.

2. For MTM transmitting MIDI data – do not transmit MIDI effects.

To program press MIDI from play patch.

1a. The display will show.

program MIDI: rx ch ignore :yes

Which is telling you that MTM is ignoring MIDI channel data that it is receiving (RX). IE. any data sent to MTM will be recognised irrespective of the MIDI channel it was sent on. (MIDI omni mode).
Press No to change.

Press Option

2. The display will show:

program MIDI MIDI Echo tx :yes

Which is telling you that MTM is transmitting the MIDI echo and effects as programmed in effect.
Press No to change.

PRESS OPTION

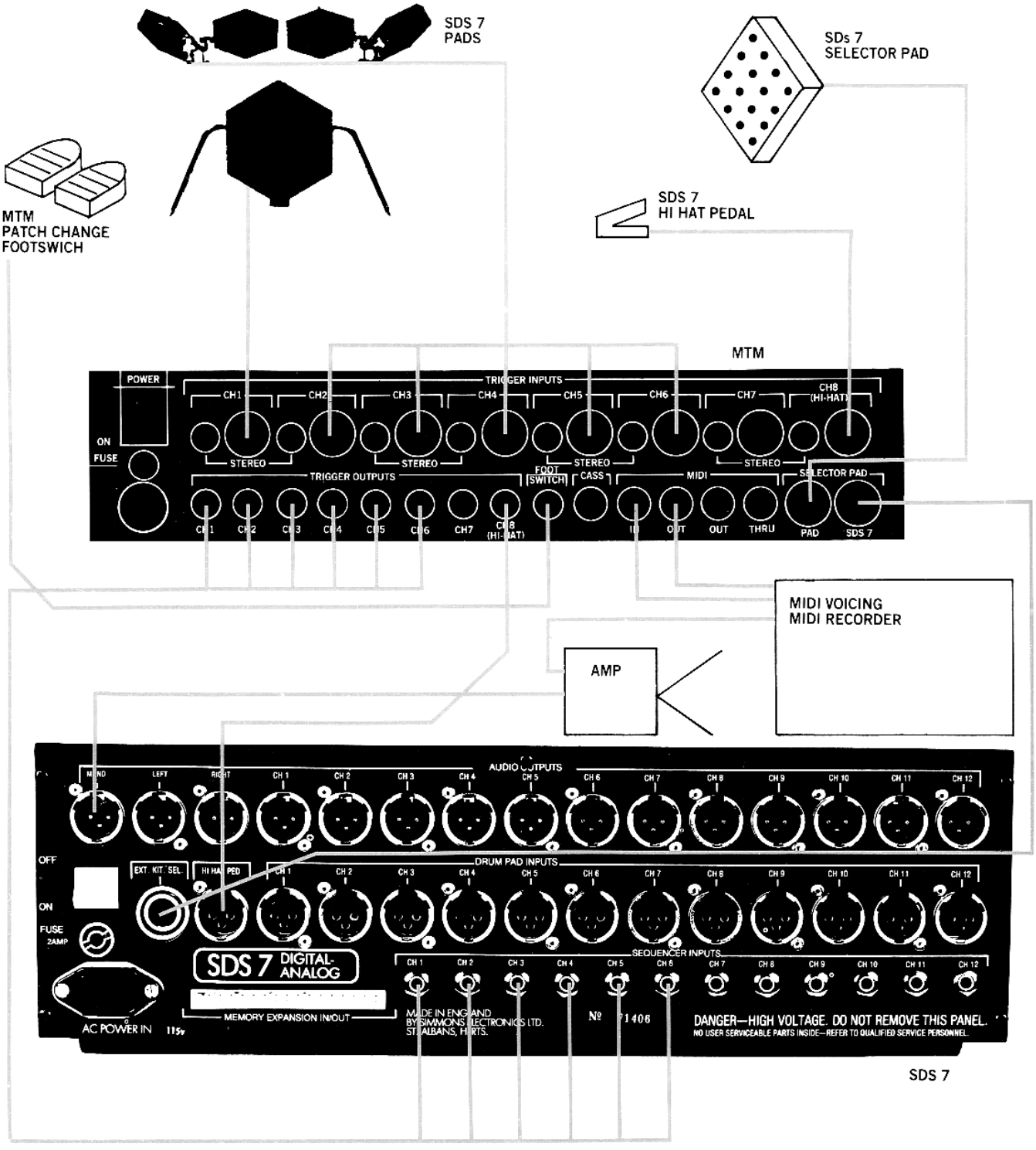
1b. The display will show:

program MIDI: MIDI echo rx :yes

This means that incoming MIDI signals will cause MTM to echo them (if a patch is being used that contains an echo effect). This is not always desirable – EG. You have played MTM into a MIDI recorder using a patch that has echo. The MIDI signals for those echoes will have been recorded by the MIDI recorder (unless you have switched echo transmit off. See above). If that recording is now played back into MTM, the incoming echoes will be echoed – causing a general mess. – Your echoes will have echoes.

In this case to play back the recording as it was played you would want to turn 'MIDI echo rx (receive) to No.'
Press No.

SDS 7 – MTM – MIDI HOOK UP



BUTTON PUSHING SUMMARY

nnnnnn = button push. **(nnnn)** = option

Press ALL/DEFAULT to toggle between Factory and user patches. (Factory is denoted by 'F' infront of patch number.)

Press **(+)/(−)** to change patch number.
Press **process.** or **route** or **effect** then press **(+)/(−)** to change process/route/effect number.

EDIT PATCH
Press **patch/edit.** press **(+)/(−)** to change patch number.
Press **process/(+)/(−)** to change process number.
Press **route/(+)/(−)** to change route number.
Press **effect/(+)/(−)** to change effect number.
Press **store** (press **(+)/(−)** to change to new patch number)/**store** to save edit.
Press **edit** to return to play patch.
Press **MIDI chan/prog (+)/(−)** to change MIDI chan number for prog change.
Press **MIDI prog/(+)/(−)** to change MIDI program change number.
Press **SDS 7 SEL PAD/(+)/(−)** to change selector pad number.

PLAY SEQUENCES (OF PATCHES).
Press **sequence** press keypad **0** to **9** to select sequence number.
Press **(+)** to move forwards or **(−)** to move backwards through sequence. (footswitch +/−).

EDIT SEQUENCE (OF PATCHES).
Press **sequence/edit.** press **(+)/(−)** to move through sequence.
Press **option** to select overwrite. (press keypad **0** to **9** to enter new data/press **ALL/DEFAULT** to toggle between factory and user patches.)
Press **option** to select insert. (press keypad **0** to **9** to enter new data/press **ALL/DEFAULT** to toggle between factory and user patches.)
Press **option** to select delete. (press keypad **0** to **9** to delete old data/press **ALL/DEFAULT** to clear complete sequence.)
Press **store ((+)/(−)** or keypad to change to new sequence number)/**store** – to save edit.
Press **edit** to return to play sequence.

PLAY ROUTE
Press **route.** Press **(+)/(−)** to change route number.

EDIT ROUTE
Press **route/edit.** Press **(+)/(−)** to change route number.
Press **trigin ((+)/(−)** to change trigin).
Press **MIDI Chan/note** to change MIDI chan for this trigin.
Press **MIDI note** to change MIDI note for this trigin.
Press **trigout** to change trig out for this trigin.
Press **Keypad 1-8** to change current trigger.
Press **Keypad '0'** to toggle note number, chromatic note name.
Press **ALL/DEFAULT** to make current parameter apply to all trigger channels.
Press **Store ((+)/(−)** to change to new route number)/**Store** to save edit.
Press **Edit** to return to play route.

BUTTON PUSHING EFFECT

BUTTON PUSHING EFFECT
Press **effect** to enter play effect.
Press **edit** to enter edit effect.
Use + / – to select effect number.
Press **trig in, MIDI note,** or **sequence.**
Alter appropriate parameters.
Press **edit** to abort, or **store,** number, **store** to save.

EDIT EFFECT – TRIG IN
Press + / – to select echo/seq type.
Press **option** to select start threshold, time, decay and number parameters.
Press + / – to change data.
Press keypad 1 to 8 to choose new trigger channel.
Press **MIDI note, sequence, edit** or **store** as appropriate.

EDIT EFFECT – EFFECT SEQUENCE
Press + / – to change sequence tempo.
Press **option** to edit sequence position, note offset and note length.
Press **trigger in, MIDI note, option, edit, store** as appropriate.

OUTPUT EFFECTS – MIDI NOTE
Press + / – to change output effect type.
Press **option** to select note or chord.
Press + / – to change data.
Press **'0'** to see MIDI note number/chromatic name whilst displaying MIDI note.
Press **'0'** to choose up or down chord whilst displaying chord names.
Press **trig in, sequence, edit, store** as appropriate.
At any time press **ALL/DEFAULT** to make displayed parameter apply to all trigger channels.

BUTTON PUSHING – PROCESS

From play process.

EDIT PROCESS

Press **process/edit.** Press **(+)/(–)** to change process number.
Press **trig in** (press **(+)/(–)** or keypad **1** to **8** to change channel) to select input processes.

Threshold

input gain	Press **threshold** to select input gain. Press **(+)/(–)** to toggle between mic and line level. Press keypad **1** to **8** to choose another channel (press **all** to change all channels to displayed setting.)
abs thresh.	Press **option** to select absolute threshold. Press **(+)/(–)** to change value. (Press **1** to **8** to change channel. press **all** to change all channels to displayed setting.)
% above prev.	Press **option** to select % above previous. Press **(+)/(–)** to change value. (Press keypad **1** to **8** to change channel. press **all** to change all channels to displayed settings.)
% above chan.	Press **option** to select % above channel. Press **(+)/(–)** to select channel to reference. Press **option** to move cursor under % value. press **(+)/(–)** to change value.

Holdoff

hold off — Press **holdoff** to select holdoff time. Press **(+)/(–)** to change holdoff time. (Press keypad **1** to **8** to change channel. Press **all** to change all channels to displayed settings.)

dyn hold off — Press **option** to select dynamic holdoff. Press **(+)/(–)** to change dynamic holdoff time. (Press keypad **1** to **8** to change channel. Press **all** to change all channels to displayed settings.)

Compress/expand

com. ex. curves. — Press **compress/expand** to select. Press **(+)/(–)** to choose curve (1 to 12). (Press keypad **1** to **8 to change** channel. Press **all** to use selected curve on all channels.)

min output. — Press **option** to select minimum output. Press **(+)/(–)** to change minimum output value. (Press keypad **1** to **8** to select channel. Press **all** to use displayed value for all channels.)

Trigger out pulse width

Press **trig in** to select trigger out processes. Press **(+)/(–)** or keypad **1** to **8** to select channel.

trig out width. — Press **width** to move cursor under width value. Press **(+)/(–)** to change value. (Press keypad **1** to **8** to change channel. Press **all** to change all channels to displayed value.)

trig out dyn width. — Press **option** to display dynamic pulse width. Press **(+)/(–)** to change value. (Press keypad **1** to **8** to change channel. Press **all** to change all channels to displayed value.)

MIDI in

MIDI abs thresh. — Press **MIDI chan/note** to select absolute threshold. Press **(+)/(–)** to change absolute threshold.

MIDI comp/ exp. — Press **compress/expand** to select MIDI compress/expand. Press **(+)/(–)** to choose dynamic curve for MIDI in.

MIDI min. output. — Press **option** to select minimum output for MIDI. Press **(+)/(–)** to change minimum output for MIDI.

MIDI out.

MIDI width — Press **MIDI note** to select first MIDI out process – fixed width. Press **(+)/(–)** to change fixed width value.

MIDI dyn width — Press **option** to select dynamic width. Press **(+)/(–)** to change MIDI dynamic pulse width.

At any time throughout previous button pushing –

Store — Press **store** ((**+**)/(**–**) or keypad to change process number). press **store** to save edited process. Press **edit** to return to play process and forget edit.

HAVING PROBLEMS – EXPLANATION OF OFTEN OCCURRING PROBLEMS

1. **MTM as an interface between pads and analogue drum voice.**

You have connected pads to MTM inputs either via XLR connectors or jack connectors. You have connected MTM trigger outputs to sequencer inputs or pads inputs on the voicing. You are striking the pads but the voices are not making any noise.

First of all plug one of your pads directly into the voicing unit, hit the pad and check that you can hear sound in this configuration. This will check out the power supply to your voicing unit, the pad and your amplification chain. Plug the pad back into MTM.

Check that MTM's 8 trigger sensitivity controls are turned to maximum. Secondly check that the trigger out control on the extreme left hand side of the unit is turned fully up.

Switch MTM to play patch factory 01. To do this, press Edit until you exit Edit (if you are in Edit). Press patch until 'play patch' appears and use the – / + buttons until F01 appears on the screen. MTM will then be playing patch F01. Failing that, a panic measure would be to switch MTM off and on again, as the 'power up state' is 'play patch F01'.

Patch F01 is matched to Simmons' jack pads. If you are using Simmons' XLR type pads MTM will not be creating a large enough trigger, so press the +1 button until the display says play patch F05. This patch is matched to XLR type Simmons' pads.

MTM as interface between pads and MIDI voicing

Follow the above procedure to get MTM into a known state with the trig-in senstitivities turned to maximum. Select factory patch 1, 5, 9, or 13 depending upon your setup. ie whether you have mono jack pads, normal XLR pads, stereo jack pads, or a kit set up with mono jacks. Check the following:

1. Is MTM 'MIDI out' connected to the 'MIDI in' socket of the voicing?
2. For all of the above MTM patches, MTM is transmitting data on MIDI channel one. – Check to see if the voicing is receiving on MIDI channel one.
3. Check to see if the voicing unit has MIDI receive switched on.
4. If the receiving voicing instrument is a synthesizer, does its' current patch have a fast attack and long release? This is tested by striking the keys of the instrument very sharply. Release the key very quickly, if you get no sound or just a short click doing this, try different patches on the voicing unit.
5. If the instrument is a drum box, check to see that both MTM and the drum box are using corresponding MIDI note numbers. Look on page 12 of this manual, at the route charts for route 1 – this is used in MTM factory patches 1, 5, 9 and 13. If you are using factory route 1, channel 1 will be transmitting MIDI note number 48 (or C4).

channel two will be transmitting MIDI note 52 or (E4),
channel three = MIDI note 55,
channel four = MIDI note 59,
channel five = MIDI note 60,
channel six = MIDI note 64,
channel seven = MIDI note 67,
channel eight = MIDI note 71.

Of course all these MIDI notes can be changed by tailoring an MTM route to suit the receiving drum box, or alternatively configuring the drum box to suit these numbers.

Is channel 8 behaving strangely? It may be that channel 8 has been configured to match an SDS 7 high hat pedal and you are attempting to use it as a normal triggering type channel like channels 1 to 7. Check to see that the SDS 7 option for channel eight is switched off. To do this turn to SDS 7 high hat.

MTM SPECIFICATION: ELECTRICAL, MECHANICAL SPECIFICATION

POWER REQUIREMENTS:

240/220.
110/100. volts internally selected AC volts, 20va.

Single board computer controlled trigger MIDI trigger interface.

Pad trigger voltage – 5 volt maximum.

Sequence trigger voltage – 15 volt maximum.

Trigger output voltage – 15 volt maximum.

Processor type – 8031.

Programme ROM 24K bytes.

User battery backed RAM 8K bytes.

Cassette dump frequence, 1200/2400 hz.

MIDI – in/through/2 x out. – note assign/voice assign/dynamic control.

Electronics dimensions including feet and rack mounting ears, 484 x 245 x 100mm. (90mm without feet).

MTM COMPRISES:

MTM Electronics
MTM Users Manual
MTM Guarantee Card
8 jack-jack leads
2 MIDI cables
Dual footswitch
Cassette dump/load lead

MTM ACCESSORIES:

SDS 7 Selector Pad Cable – Connects MTM to SDS 7 Selector Pad Input.

Specification subject to change without notice.

SIMMONS

Simmons Electronics Limited
Alban Park, Hatfield Road, St Albans, Herts AL4 0JH
Tel: (0727) 36191 (5 lines). Telex: 291326 HEXDRM G

Thanks for visiting
http://www.simmonsmuseum.com