Mackie Control Protocol

The results of 3 weeks reverse-engineering and development of a translation from MCP to OSC

by Nicolas Jarnoux

10/16/16

Content

Preface

As you know, Mackie Control protocol is an almost twenty years old protocol. Everyone knows it, from equipment manufacturers to final users but Mackie never published an implementation charts. Unfortunatly, I don't own any MCP-able control surface, it was my principal motivation to do this project (to control any DAW with an iPad or any in OSC). This document is part of this personal project that consists in a translation from MCP [Mackie control Protocol] to a clear OSC charts, and provides the results of my researches about this protocol.

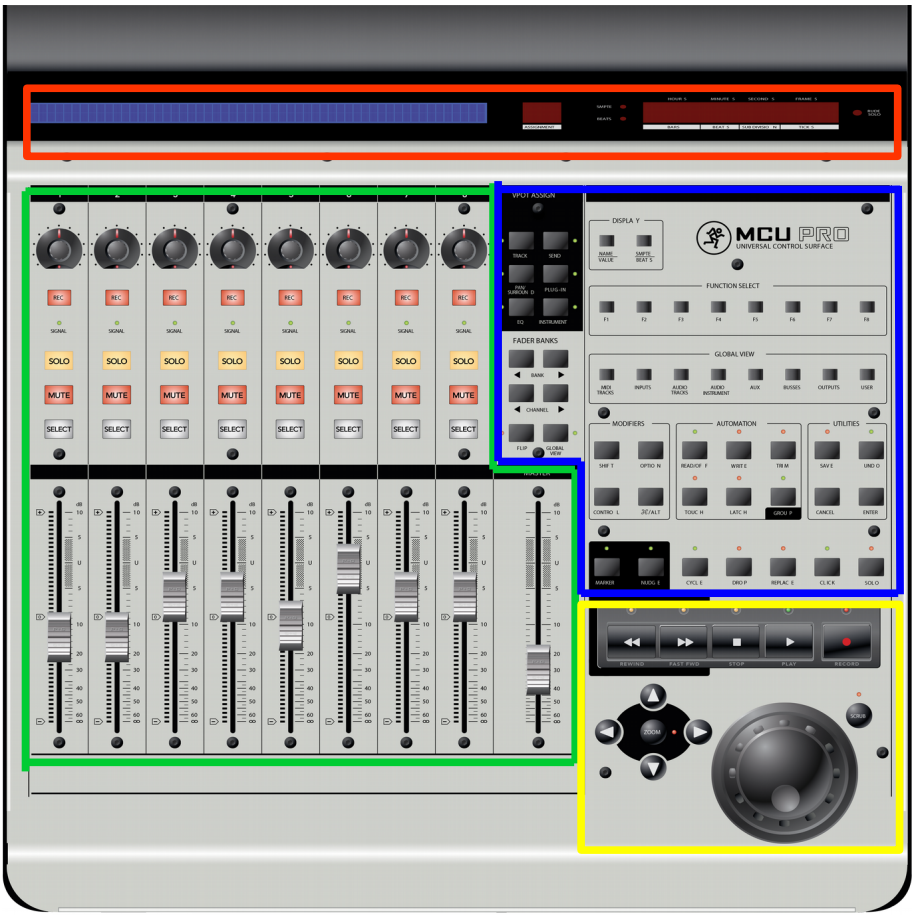
I am not a professional, just a passionate geek in sound and IT. That’s why the results might be false or misunderstood but I think I came to a fair outcome.

I decided to publish this document to make up for the lack of technical documentation on the internet.

Just before I start, I want to thanks *theageman* and two other *unknown authors* for their documents that really helped me and by the way, as a french guy, I sincerely apologize for the english mistakes I could do here. However my girlfriend Julie Lacave accepted (very nicely) to help me!

Anatomy of a Mackie Control Universal

I chose to work on the Mackie Control Universal Pro (2nd gen). In this part I will describe how controls are organized on the surface.



- The red zone is the display zone
- The green zone is the faders channels
- The blue zone is the functions buttons zone
- The yellow zone is the transport zone

I will talk about each zone one by one describing how it works, what it sends, what it listens.

General behavior

I will spend more time to explain particular functions / messages. But the behavior of buttons and leds is always the same: bang notes. To bang a note consists in send a Note On midi message and immediately the corresponding Note Off, allow midi APIs to not bet overloaded with too many Note On simultaneously.

Each buttons are associated to a midi note on the channel 1. Each leds are associated to the same note that their corresponding button.

The status of the button (pressed, released) or the leds (on, off) are transmit with the velocity value.

When I press a button, MCU[1] bangs the corresponding note with a velocity higher than 0, often and recommended 127, the highest one. When I release the button, MCU bangs the corresponding note with a velocity strictly equal to 0.

Same rule with leds, DAW[2] will bang the corresponding note with a velocity higher than 0 to turn it on and with a velocity equal to 0 to turn it off.

1 MCU: Mackie Control Universal or any control surface compatible with MCP
2 DAW: Digital Audio Worstation, any software compatible with MCP

The display zone

Represented in red on the schematics above, it is composed of one 2*56 characters LCD screen, one assignment display (2 digits, 7 segments each), one timecode display (10 digits, 7 segments + dot each) and 3 leds.

This part is only listening to what the DAW is sending, it never interacts with in other way.

LCD screen

It is composed of two lines of 56 characters each, 112 characters in total, and reacts to a System Exclusive midi message sent by the DAW. Here is its content, all values are in hexadecimal:

```
F0 00 00 66 14 12 XX CC CC [..] F7
```

In this example:

- F0 and F7 are respectively start and end of Sysex messages
- 00 00 66 is the manufacturer's ID (Makie ID in this case)
- 14 is the product's ID (MCU ID in the case)
- 12 says that it talks to the LCD screen
- XX is the first character position. This is a value between 00 and 6F (111 in decimal) included.
  - 00 is top-left character
  - 37 is the top-right character. (55 in decimal)
  - 38 is the bottom-left character (56 in decimal)
  - 6F is the bottom-right character (111 in decimal)
- CC CC [..] is the list of characters that have to be changed. The first character will replace the XXth char, the second will replace the XXth+1 char, etc... Characters set is describe at appendix, page 17.

For example, if I want to fill the ten first characters of the second line of the LCD with asterisk (*) and let the rest of the line as it is, I will send this message:

```
F0 00 00 66 14 12 38 2A 2A 2A 2A 2A 2A 2A 2A 2A F0
```

38: point to the first character of the second line.

2A: the ten characters I want to change starting from the **38** position.

The Assignment display

Aaaaaaaand this is the first crash, I have no f*cking idea of what this display does. If you have informations concerning that, please send me an e-mail and I will correct this part.

The SMPTE and BEATS leds

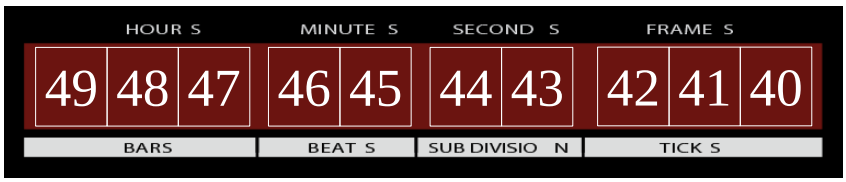
These two ones are listening to note bangs from the DAW.

SMPTE led corresponding to: channel 1, note F8 (dec: 113, hex: 71)

BEATS led corresponding to: channel 1, note F#8 (dec: 114, hex: 72)

Timecode display

This display is composed of 10 digits of 7 segment + dot each. The datas displayed are sent by the DAW in Control Change message, on the 16th channel. Control Change Ids are from 64 (hex: 40) for right digit to 73 (hex: 49) for the left digit.



The control change's Value field indicate what to display. This octet is divided in two quartets. The first one (MSB) is the control quartet, the second one (LSB) is the value quartet.

Here is the value octet with its functions:

Control				Value			
0	1	1	1	0	0	0	1
Unused, always at 0	Dot bit 1=dot on 0=dot off	Unused, always at 1	General on/off bit 1=digit on 0=digit off	Value to display from 0 (0000) to 9 (1001). Maybe A to F are accepted but I have not seen the case...			

In this example, I will display the number "1" and turn the digit's dot on. Other examples:

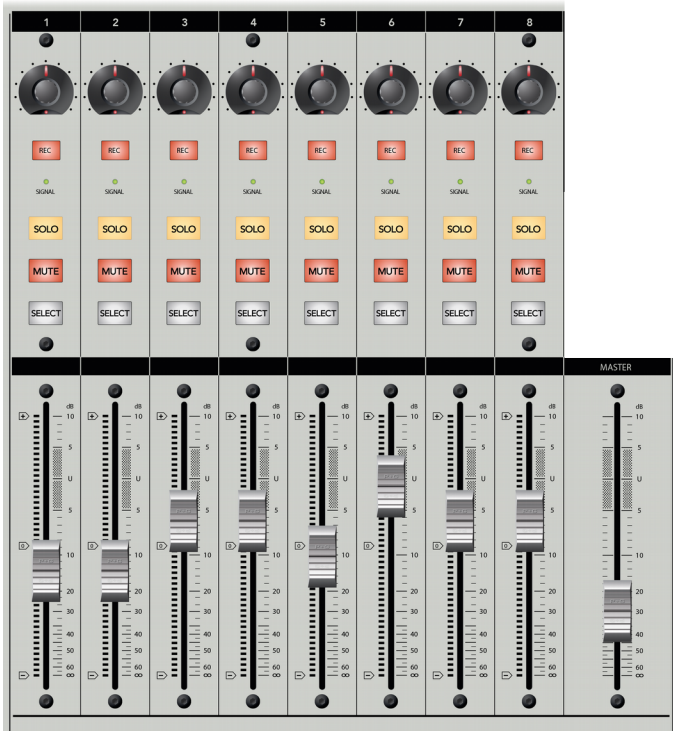
- Channel 16, CC 72 (hex: 48), Value 32 (hex: 20): turn off the middle digit of BARS
- Channel 16, CC 67 (hex: 43), Value 51 (hex: 33): set the first SUB DIVISION digit to 3 without dot
- Channel 16, CC 69 (hex: 45), Value 121 (hex: 79): set the first BEATS digit to 9 with dot on

Rude solo led

Same thing that Assignment display, I don't have any information about that...

The Mixer zone

This is the principal par of the MCU, the mixer is composed of 8 complete channels with virtual clickable pot with infinite rotation, 4 buttons for select, mute, solo and armed rec, a 12 leds VU meter (depending of the model) and a motorized fader. On the right the Master motorized fader.



Virtual pot

A virtual pot (vPot) is composed of a clickable knob with infinite rotation, 11 Leds ring, 1 led under the knob.

Rotating the vPot

When a vPot is rotated, the MCU sends a CC (Control Change) message:

- Channel 1
- CC 16 (hex: 10) to 23 (hex: 17), 16 is the first channel, 23 is the eighth channel.

When the pot is rotated clockwise, the MCU sends a value equal to 1, when it is rotated counter clockwise, the MCU sends a value equal to 65 (hex: 41). I can't test any other values but the 7th bit of the octet (64 value in decimal) seems to be a "sign indicator". Maybe the MCU sends a value between 1 and 63 (hex: 3F) when the pot is turned clockwise and between 65 (hex: 41) and 127 (hex: 7F) when the pot is turned counter clockwise in function of the rotation speed.

Clicking the vPot

When a vPot is clicked, the MCU bangs a midi note where the note is depending of the vPot's channel and the velocity hold the switch status.

- Channel 1
- Note from G#1 (dec: 32, hex: 20) to D#2 (dec: 39, hex: 27) for channel 1 to 8.
- Velocity 127 (hex: 7F) when switch is pressed, 0 when it is released.

Leds ring

Each vPot is surrounded by 12 leds. 11 of them indicate the actual pot value and 1 smaller led is just under the pot. All these leds are controlled by the DAW using Control Change message.

- Channel 1
- CC from 48 (hex: 30) to 55 (hex: 37) for channel 1 to 8

The value octet is separated in 2 quartets. The MSB one is used to control the display mode and the LSB one is used for led values.

```
0 S M M  V V V V
```

In the control quartet, the two LSB bits (M M in the example) encode the four display modes (from 00 to 11 in binary), the third bit (S) commands the status of the smaller led under the pot and the fourth bit is not used and always at 0.

The value quartet (VVVV) encodes from 0 to B the status of the 11 leds around the pot like this:

	Mode 00 (hex: 0)	Mode 01 (hex: 1)	Mode 10 (hex: 2)	Mode 11 (hex: 3)
0	`-----------`	`-----------`	`-----------`	`-----------`
1	`*----------`	`******-----`	`*----------`	`-----*-----`
2	`-*---------`	`-*****-----`	`**---------`	`----***----`
3	`--*--------`	`--****-----`	`***--------`	`---*****---`
4	`---*-------`	`---***-----`	`****-------`	`--*******--`
5	`----*------`	`----**-----`	`*****------`	`-*********-`
6	`-----*-----`	`-----*-----`	`******-----`	`**********`
7	`------*----`	`-----**----`	`*******----`	`**********`
8	`-------*---`	`-----***---`	`********---`	`**********`
9	`--------*--`	`-----****--`	`*********--`	`**********`
A	`---------*-`	`-----*****-`	`**********-`	`**********`
B	`----------*`	`-----******`	`**********`	`**********`

Rec / Solo / Mute / Select buttons

All these buttons operate like standard buttons, that is to say the MCU bangs a note with the highest velocity when they are pressed and bangs the same note with 0 velocity when they are released. The corresponding leds under the buttons are activated by the DAW by banging the same note that the corresponding button with highest velocity to turn it on and 0 velocity to turn it off. All notes are sent on the midi channel 1.

- from C-1 (dec: 0, hex: 0) to G-1 (dec: 7, hex: 7) for REC buttons, channel 1 to 8
- from G#-1 (dec: 8, hex: 8) to D0 (dec: 15, hex: F) for SOLO buttons, channel 1 to 8
- from E0 (dec: 16, hex: 10) to B0 (dec: 23, hex: 17) for MUTE buttons, channel 1 to 8
- from C1 (dec: 24, hex: 18) to G1 (dex: 31, hex: 1F) for SELECT buttons, channel 1 to 8

VU Meters leds

VU meter of each channel are commanded by the DAW using Channel Pressure midi message, on midi channel 1. The value octet is once again divided in 2 quartets. The MSB one represents the MCU channel number from 0 to 7 (channel 1 to 8), the LSB one encodes the 12 VU Meter leds status from 0 to 12 (hex: C).

Value	Signal	Leds
C	>= 0dB	Red (clip)
B	>= -2dB	Yellow
A	>= -4dB	Yellow
9	>= -6dB	Yellow
8	>= -8dB	Green
7	>= -10dB	Green
6	>= -14dB	Green
5	>= -20dB	Green
4	>= -30dB	Green
3	>= -40dB	Green
2	>= -50dB	Green
1	>= -60dB	Green
0	< -60dB	All leds off

Faders control

Faders work with Pitch Bend messages and banging notes. When a fader is touched, the MCU unlocks the fader motor and bangs a note on midi channel 1 to advise the DAW that the fader will move. Highest velocity when a fader is touch, 0 velocity when the fader is released

- from G#7 (dec: 104, hex: 68) to E8 (dec: 112, hex: 70) for faders 1 to 8 and master fader

When you move a fader, the MCU sends Pitch Bend messages to advise the DAW of the new fader position.

When the DAW wants to move a fader (automation or changing fader bank), it will send the same Pitch Bend messages to advice MCU of the new fader position.

Fader position is encoded on the 14 bits of the Pitch Bend value fields (from 0 to 16383 or -8192 to 8191, depends on your midi API implementation). Fader middle position is the middle value of pitch bend.

Pitch bend midi channel is from 1 to 9 for faders 1 to 8 and master fader

A complete moving fader sequence looks like that:

1. touch fader
2. move fader
3. move fader
4. ...
5. move fader
6. release fader

Functions zone

The functions zone is only composed of buttons, some of them with a led. This part follow the general behavior described above (the MCU bangs notes for pressing/releasing action, the DAW bangs notes to turn on/off corresponding leds). A table retracing all buttons in correspondence with midi notes is in appendix, page 13.

Transport zone

This zone is composed of transport and zoom controls and a scrub wheel. All buttons act as others and there corresponding notes are related in appendix, page 13.

Scrub wheel

The scrub wheel acts as the vPot rotation(page 8).

When the scrub wheel is rotated, the MCU sends a CC (Control Change) message:

- Channel 1
- CC 60

When the wheel is rotated clockwise, the MCU sends a value equal to 1, when it is rotated counter clockwise, the MCU sends a value equal to 65 (hex: 41). Same comment that for vPots, maybe the MCU sends values in function of the rotation speed but no way to test it...

I. Table of controls and corresponding midi values

Item	Note		Velocity On	Velocity Off	Channel
Rec 1	C-1	Dec: 0, hex: 0	127 (7F)	0	1
Rec 2	C#-1	Dec: 1, hex: 1			
Rec 3	D-1	Dec: 2, hex: 2			
Rec 4	D#-1	Dec: 3, hex: 3			
Rec 5	E-1	Dec: 4, hex: 4			
Rec 6	F-1	Dec: 5, hex: 5			
Rec 7	F#-1	Dec: 6, hex: 6			
Rec 8	G-1	Dec: 7, hex: 7			
Solo 1	G#-1	Dec: 8, hex: 8	127 (7F)	0	1
Solo 2	A-1	Dec: 9, hex: 9			
Solo 3	A#-1	Dec: 10, hex: A			
Solo 4	B-1	Dec: 11, hex: B			
Solo 5	C0	Dec: 12, hex: C			
Solo 6	C#0	Dec: 13, hex: D			
Solo 7	D0	Dec: 14, hex: E			
Solo 8	D#0	Dec: 15, hex: F			
Mute 1	E0	Dec: 16, hex: 10	127 (7F)	0	1
Mute 2	F0	Dec: 17, hex: 11			
Mute 3	F#0	Dec: 18, hex: 12			
Mute 4	G0	Dec: 19, hex: 13			
Mute 5	G#0	Dec: 20, hex: 14			
Mute 6	A0	Dec: 21, hex: 15			
Mute 7	A#0	Dec: 22, hex: 16			
Mute 8	B0	Dec: 23, hex: 17			
Sel 1	C1	Dec: 24, hex: 18	127 (7F)	0	1
Sel 2	C#1	Dec: 25, hex: 19			
Sel 3	D1	Dec: 26, hex: 1A			
Sel 4	D#1	Dec: 27, hex: 1B			
Sel 5	E1	Dec: 28, hex: 1C			
Sel 6	F1	Dec: 29, hex: 1D			
Sel 7	F#1	Dec: 30, hex: 1E			
Sel 8	G1	Dec: 31, hex: 1F			
Vpot switch 1	G#1	Dec: 32, hex: 20	127 (7F)	0	1
Vpot switch 2	A1	Dec: 33, hex: 21			
Vpot switch 3	A#1	Dec: 34, hex: 22			
Vpot switch 4	B1	Dec: 35, hex: 23			
Vpot switch 5	C2	Dec: 36, hex: 24			
Vpot switch 6	C#2	Dec: 37, hex: 25			
Vpot switch 7	D2	Dec: 38, hex: 26			
Vpot switch 8	D#2	Dec: 39, hex: 27			

Item	Note		Velocity On	Velocity Off	Channel
Assign Track	E2	Dec: 40, hex: 28	127 (7F)	0	1
Assign Send	F2	Dec: 41, hex: 29			
Assign Pan/Surround	F#2	Dec: 42, hex: 2A			
Assign Plug-in	G2	Dec: 43, hex: 2B			
Assign EQ	G#2	Dec: 44, hex: 2C			
Assign Instrument	A2	Dec: 45, hex: 2D			
Bank Left	A#2	Dec: 46, hex: 2E	127 (7F)	0	1
Bank Right	B2	Dec: 47, hex: 2F			
Channel Left	C3	Dec: 48, hex: 30			
Channel Right	C#3	Dec: 49, hex: 31			
Flip	D3	Dec: 50, hex: 32			
Global	D#3	Dec: 51, hex: 33			
Name / Value	E3	Dec: 52, hex: 34	127 (7F)	0	1
SMPTE / BEATS	F3	Dec: 53, hex: 35			
F1	F#3	Dec: 54, hex: 36	127 (7F)	0	1
F2	G3	Dec: 55, hex: 37			
F3	G#3	Dec: 56, hex: 38			
F4	A3	Dec: 57, hex: 39			
F5	A#3	Dec: 58, hex: 3A			
F6	B3	Dec: 59, hex: 3B			
F7	C4	Dec: 60, hex: 3C			
F8	C#4	Dec: 61, hex: 3D			
MIDI Tracks	D4	Dec: 62, hex: 3E	127 (7F)	0	1
Inputs	D#4	Dec: 63, hex: 3F			
Audio Tracks	E4	Dec: 64, hex: 40			
Audio Intruments	F4	Dec: 65, hex: 41			
Aux	F#4	Dec: 66, hex: 42			
Busses	G4	Dec: 67, hex: 43			
Outputs	G#4	Dec: 68, hex: 44			
User	A4	Dec: 69, hex: 45			
Shift	A#4	Dec: 70, hex: 46	127 (7F)	0	1
Option	B4	Dec: 71, hex: 47			
Control	C5	Dec: 72, hex: 48			
Alt	C#5	Dec: 73, hex: 49			
Read/Off	D5	Dec: 74, hex: 4A	127 (7F)	0	1
Write	D#5	Dec: 75, hex: 4B			
Trim	E5	Dec: 76, hex: 4C			
Touch	F5	Dec: 77, hex: 4D			
Latch	F#5	Dec: 78, hex: 4E			
Group	G5	Dec: 79, hex: 4F			

Item	Note		Velocity On	Velocity Off	Channel
Save	G#5	Dec: 80 , hex: 50	127 (7F)	0	1
Undo	A5	Dec: 81 , hex: 51			
Cancel	A#5	Dec: 82 , hex: 52			
Enter	B5	Dec: 83 , hex: 53			
Markers	C6	Dec: 84 , hex: 54	127 (7F)	0	1
Nudge	C#6	Dec: 85 , hex: 55			
Cycle	D6	Dec: 86 , hex: 56			
Drop	D#6	Dec: 87 , hex: 57			
Replace	E6	Dec: 88 , hex: 58			
Click	F6	Dec: 89 , hex: 59			
Solo	F#6	Dec: 90 , hex: 5A			
Rewind	G6	Dec: 91 , hex: 5B	127 (7F)	0	1
Forward	G#6	Dec: 92 , hex: 5C			
Stop	A6	Dec: 93 , hex: 5D			
Play	A#6	Dec: 94 , hex: 5E			
Record	B6	Dec: 95 , hex: 5F			
Up	C7	Dec: 96 , hex: 60	127 (7F)	0	1
Down	C#7	Dec: 97 , hex: 61			
Scrub	D7	Dec: 98 , hex: 62			
Zoom	D#7	Dec: 99 , hex: 63			
Left	E7	Dec: 100, hex: 64			
Right	F7	Dec: 101, hex: 65			
?	F#7	Dec: 102, hex: 66	?	?	1
	G7	Dec: 103, hex: 67			
	G#7	Dec: 104, hex: 68			
	A7	Dec: 105, hex: 69			
	A#7	Dec: 106, hex: 6A			
	B7	Dec: 107, hex: 6B			
	C8	Dec: 108, hex: 6C			
	C#8	Dec: 109, hex: 6D			
	D8	Dec: 110, hex: 6E			
	D#8	Dec: 111, hex: 6F			
	E8	Dec: 112, hex: 70			
SMPTE Led	F8	Dec: 113, hex: 71	127 (7F)	0	1
BEATS Led	F#8	Dec: 114, hex: 72			
?	G8	Dec: 115, hex: 73	?	?	1
	G#8	Dec: 116, hex: 74			
	A8	Dec: 117, hex: 75			
	A#8	Dec: 118, hex: 76			
	B8	Dec: 119, hex: 77			

Item	Control Change	Value CW	Value CCW	Channel
Vpot 1 rotation	Dec: 16, hex: 10	1	65 (hex: 41)	1
Vpot 2 rotation	Dec: 17, hex: 11			
Vpot 3 rotation	Dec: 18, hex: 12			
Vpot 4 rotation	Dec: 19, hex: 13			
Vpot 5 rotation	Dec: 20, hex: 14			
Vpot 6 rotation	Dec: 21, hex: 15			
Vpot 7 rotation	Dec: 22, hex: 16			
Vpot 8 rotation	Dec: 23, hex: 17			
Item	**Control Change**	**Value**		**Channel**
Vpot 1 leds	Dec: 48, hex: 30	Explain at page 9		1
Vpot 2 leds	Dec: 49, hex: 31			
Vpot 3 leds	Dec: 50, hex: 32			
Vpot 4 leds	Dec: 51, hex: 33			
Vpot 5 leds	Dec: 52, hex: 34			
Vpot 6 leds	Dec: 53, hex: 35			
Vpot 7 leds	Dec: 54, hex: 36			
Vpot 8 leds	Dec: 55, hex: 37			
Item	**Control Change**	**Value CW**	**Value CCW**	**Channel**
Scrub wheel	Dec: 60, hex: 3C	1	65 (hex: 41)	1
Item	**Control Change**	**Value**		**Channel**
TC Digit 1	Dec: 64, hex: 40	Explain at page 7		15
TC Digit 2	Dec: 65, hex: 41			
TC Digit 3	Dec: 66, hex: 42			
TC Digit 4	Dec: 67, hex: 43			
TC Digit 5	Dec: 68, hex: 44			
TC Digit 6	Dec: 69, hex: 45			
TC Digit 7	Dec: 70, hex: 46			
TC Digit 8	Dec: 71, hex: 47			
TC Digit 9	Dec: 72, hex: 48			
TC Digit 10	Dec: 73, hex: 49			
Item	**Channel Pressure**	**Value**		**Channel**
Vu Meter	/	Explain at page 10		1
Item	**Pitch Bend**	**Value**		**Channel**
Fader 1	/	Explain at page 11		1
Fader 2	/			2
Fader 3	/			3
Fader 4	/			4
Fader 5	/			5
Fader 6	/			6
Fader 7	/			7
Fader 8	/			8
Master	/			9

II. Character set of Mackie Control

Actually, it is a classic ASCII table:

Hex	Display		Hex	Display		Hex	Display		Hex	Display
00			20	space		40	@		60	`
01			21	!		41	A		61	a
02			22	"		42	B		62	b
03			23	#		43	C		63	c
04			24	$		44	D		64	d
05			25	%		45	E		65	e
06			26	&		46	F		66	f
07			27	'		47	G		67	g
08			28	(		48	H		68	h
09			29	)		49	I		69	i
0A			2A	*		4A	J		6A	j
0B			2B	+		4B	K		6B	k
0C			2C	,		4C	L		6C	l
0D			2D	-		4D	M		6D	m
0E			2E	.		4E	N		6E	n
0F			2F	/		4F	O		6F	o
10			30	0		50	P		70	p
11			31	1		51	Q		71	q
12			32	2		52	R		72	r
13			33	3		53	S		73	s
14			34	4		54	T		74	t
15			35	5		55	U		75	u
16			36	6		56	V		76	v
17			37	7		57	W		77	w
18			38	8		58	X		78	x
19			39	9		59	Y		79	y
1A			3A	:		5A	Z		7A	z
1B			3B	;		5B	[		7B	{
1C			3C	<		5C	\		7C	|
1D			3D	=		5D	]		7D	}
1E			3E	>		5E	^		7E	~
1F			3F	?		5F	_		7F	

III. Summary of midi message format

Message	Status Byte D7..D0	Data1 Byte D7..D0	Data2 Byte D7..D0	Explanation
Note Off	1000cccc	0nnnnnnn	0vvvvvvv	C: Channel N: Note V: Velocity
Note On	1001cccc	0nnnnnnn	0vvvvvvv	C: Channel N: Note V: Velocity
AfterTouch Key pressure	1010cccc	0nnnnnnn	0vvvvvvv	C: Channel N: Note V: Velocity
Control Change	1011cccc	0nnnnnnn	0vvvvvvv	C: Channel N: Ctrl number V: Value
Program Change	1100cccc	0ppppppp	/	C: Channel P: Program
AfterTouch Chan Pressure	1101cccc	0vvvvvvv	/	C: Channel V: Value
Pitch Bend	1110cccc	0lllllll	0mmmmmmm	C: Channel L: LSB bits M: MSB bits
Sysex start	11110000	/	/	/
Sysex end	11110111	/	/	/