Literature review for audio control systems

Brent Shaw - G09S2665

27/05/2013

Abstract

This literature review aims to discuss various aspects of audio control systems. Through the course of this literature review, discrete field topics of study are brought together to contribute to the understanding of what is required to develop a comprehensive audio control system.

Contents

1 Introduction

The ability to control audio mixing consoles remotely has, in the past, primarily been reserved for expensive proprietary systems or has involved expensive modification. Mixing consoles often require placement that is out of the way due to their size and the number of inputs and outputs. Remote control of a mixing console would enable audio engineers to move around more freely and possibly allow for control of multiple consoles simultaneously.

A successful control system would be capable of controlling many devices from different manufacturers. It would be capable of exerting full control over mixing consoles, effects units and synthesisers. For the system to be as effective as possible, the control application would need to operate from a handheld, mobile device which the user interacts with naturally.

This literature review will explore various methods of control for mixing consoles, as well as methods of remote control that have been employed in the past for other devices.

2 MIDI: Musical Instrument Digital Interface

2.1 History

The advent of synthesizers revolutionized the music industry in the 1960's and 1970's with musicians being capable of producing a wide range of new sound. But an early issue that arose with synthesizers was that many could play only one note at a time. This had obvious limitations and manufacturers attempted to rectify this by implementing control voltage systems that allowed for a keyboard to control a synthesizer. This allowed for more than one note to be played at a time, but resulted in synthesizers becoming covered in control cables and setups becoming increasingly complicated. (Rothstein, 1995)

Dave Smith and Chet Wood, engineers working with Sequential Circuits, started work on a new protocol that would allow for the interoperability of synthesizers across different manufacturers. In 1981 they proposed their USI (Universal Synthesizer Interface) protocol to the Audio Engineering Society. (Huber, 2012) Manufacturers and members of the audio engineering community began working on a standard for synthesizer control, and in 1983 the MIDI 1.0 Specification was born. (Hosken, 2010)

The MIDI (Musical Instrument Digital Interface) standard was widely accepted with many new technologies at the time being released with MIDI compatibility. Today MIDI is still widely used and has been incorporated into synthesizers, discrete keyboards, effects units, guitars, mixing consoles, lighting consoles and is a key component in computer music production. (Huber, 2012) (Hosken, 2010) (Jacobs & Georghiades, 1991)

2.2 Hardware

Most MIDI devices offer a selection of three MIDI ports, namely input, output and through. The through (often written "thru") is optional and allows for the daisy-chaining of MIDI devices. Typically, The MIDI standard uses 5 pin DIN connections, but not all 5 pins are operational: only pins 2, 4 and 5 are used. (Jacobs & Georghiades, 1991)

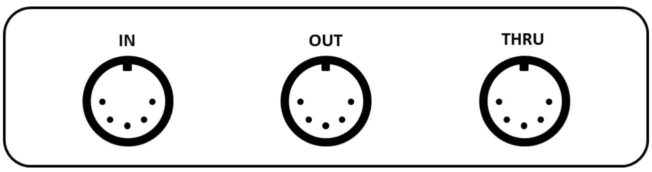

Figure 1: Example setup of MIDI connection on a MIDI device.

The MIDI specification requires that the interface operates at 31.25 kbaud in an asynchronous manner. The interface's receiver is required to be onto-isolated and an output is not allowed to feed more than one input. (Huber, 2012)

2.3 Messaging Protocol

MIDI communicates using a series of bytes, either being status bytes or data bytes. A message is constituted by an initial status bytes followed by one or more optional data bytes. There are two types of messages, namely channel messages and system messages. (DeFuria & Scacciaferro, 1990)

2.3.1 Channel Messages

MIDI devices operate on a selected MIDI channel, and so send control messages pertaining specifically to that channel, these messages are referred to as channel messages. Channel messages can further be divided into channel voice messages and channel mode messages. (DeFuria & Scacciaferro, 1990)

The former, voice messages, can be used to transmit note messages, program messages and expressive messages. Note messages are generated when a key on a keyboard controller is pressed; this is referred to as a "note-on" messages. When the depressed key is lifted, a "note-off" message is sent. (Kirk & Hunt, 1999)

The necessity for a both "on" and "off" messages is due to that fact that if the system was required to wait for the user to complete playing a note before transmission, considerable delay would be incurred. (Rothstein, 1995) When a key is pressed, the "note-on" message is sent with information such as which key was pressed (key number) and how hard (velocity). (Hosken, 2010)

Status Byte	Hex	Decimal	Data Byte 1	Data Byte 2
Note off	80-8F	128-143	Note number	Release velocity
Note on	90-9F	144-159	Note number	Attack velocity
Poly key pressure	A0-AF	160-175	Note number	Pressure value
Control change	B0-BF	176-191	Controller ID	Controller value
Program change	C0-CF	192-207	Program number	-
Channel pressure	D0-DF	208-223	Pressure value	-
Pitch wheel change	E0-EF	224-239	Pitch bend (LSB)	Pitch bend (MSB)

Figure 2: Table showing various types of channel voice messages. Adapted from "MIDI Programmer's Handbook" (DeFuria & Scacciaferro, 1990)

2.3.2 System Messages

In contrast to channel messages, MIDI systems messages broadcast to all MIDI devices. One feature of system messages is the ability to send device-specific messages know as "Exclusive Messages". System messages can also be used to communicate "Time" messages allowing for devices to be synchronised, as well as "Common" messages used for tuning and setup. (Kirk & Hunt, 1999)

2.4 Controlling with MIDI

MIDI, although not initially designed for such use, can be implemented for use in controlling other devices in the recording or production studio. Devices such as effects units and mixing desks can be controlled using MIDI keyboard controllers or other control surfaces. One distinct advantage gained through using MIDI controlled mixing consoles is that of automation. Automation allows levels to fade over a period of time without the engineer having to slowly pull down on a fader. The ability to program the change of parameter so that the user need not constantly change it, be it a level, panning or an equalisation parameter, can not only make mixing a multitrack recording a lot simpler, but can also allow for engineers to attempt newer and more intricate mixing techniques. This can all be achieved by using a MIDI sequencer to store MIDI messages and have them played back during mixdown.With the incorporation of MIDI into computers and applications, control for mixing consoles and effects can be even simpler. Users can change parameters on an on screen application and have the changes reflected on the device being controlled. These changes can be recorded, modified and re-used by employing software based sequencers. (Huber, 2012)

2.5 Limitations and the future of MIDI

Since the MIDI specification has never been revised since its release, MIDI has a few limitations to its operation and its capabilities. MIDI cables are limited (in the standard) to being up to 15m long, to avoid the risk that attenuation might result in signal loss. (Jacobs & Georghiades, 1991) This, although seeming fairly pedestrian, is an issue, as large synthesisers, despite being capable of being distanced from their keyboard controller, are restricted by this maximum length of cable. Although there are systems available that can extend this distance, and some systems that even make it possible to send MIDI point to point wirelessly, this is still a restriction of MIDI.

Another issue with MIDI is its speed. MIDI operates serially, transferring bits one after another, at what is considered a slow speed: 31.25 kbaud. This can lead to the problem of "MIDI lag". Where many complex MIDI messages and many channels are being used, MIDI messages can begin to lag due to the slow speed of transmission. This speed limit makes MIDI less reliable in high demand areas such as recording and production studios. The same lag can be experienced when many MIDI devices are daisy-chained together. (DeFuria & Scacciaferro, 1990)

Another issue with MIDI is its limit to only 16 channels. For a single device, in general, 16 channels is a fair limit, with most synthesisers only being capable of generating a limited number of voices. Unfortunately, with most setups sporting multiple MIDI devices, 16 channels is too restricting.(Kirk & Hunt, 1999)

MIDI has never been revised, and thus many of these limitations might only be due to the protocol's age. MIDI would definitely benefit from an increase in speed and with modern architecture this would not pose much of an issue. The issue with attempting to move MIDI forward would seem to lie in its widespread acceptance. So many devices already incorporate MIDI, an update to MIDI would need to remain backward compatible with existing MIDI devices.

3 Other control methods

3.1 MIDINet

3.1.1 History

MIDI was originally designed as a means of communication for keyboards and synthesizers, and its simple implementation has led to high uptake by manufacturers. Although the protocol was first used for the remote control of synthesisers and sound generation devices, the protocol quickly found uses in other areas, such as control for MIDI patchbays, effects units and mixing consoles. This means that it is possible to control many devices in a studio using MIDI. (Mosala, 1995)

Studio equipment is expensive and is often bulky, resulting in studio resources having to be tied to a single location. MIDI provides a method of controlling these resources remotely. Rhodes University created a network to provide control for these devices: the Rhodes Computer Music Network (RHOCMN). The network's purpose was to provide an efficient and cost-effective method of sharing music resources. (Mosala, 1995)

3.1.2 Concept

Since many devices have MIDI connections and it is possible to to provide computers with a MIDI interface, a network of computers running control software are able to provide control for audio devices. The MIDINet system provides a method of sending MIDI over a Local Area Network (LAN). (Mosala, 1995)

> "The MIDINet system allows the user to look at the collection of his MIDI devices as a single, unified system." (Mosala, 1995)

MIDI devices can be connected to a computer running the MIDINet system, enabling them to control devices connected to another computer on the LAN. Each computer is equipped with a network interface controller (NIC) and a MIDI interface card, these computers are referred to as MIDINet units. MIDINet units are connected to an Ethernet network. (Mosala, 1995)

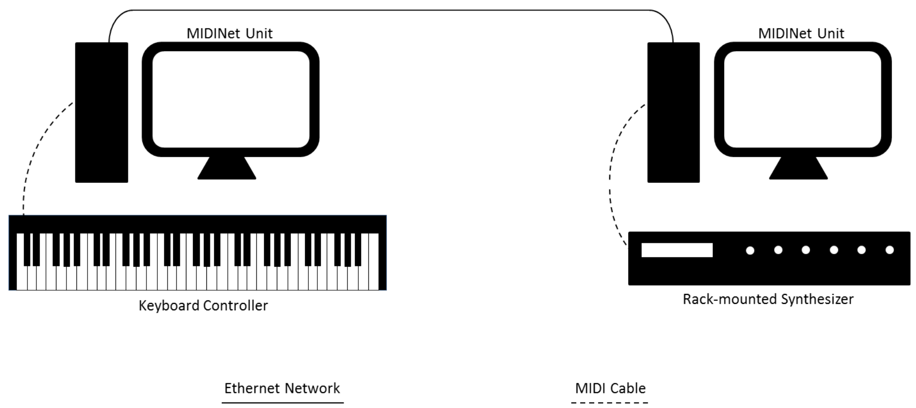

Figure 3: Example setup for a small MIDINet system.

3.1.3 Implementation

As a MIDINet unit is equipped with a MIDI interface, there will usually be at least one In and one Out port that MIDI devices can be connected to. The MIDINet system gives the ability to have many devices on a network and thus the devices must be identified by some manner. Users set device identification through the MIDINet systems menu, the information includes a MIDINet ID, a port ID, a symbolic name and a channel number. (Mosala, 1995)

MIDINet Unit ID	Port ID	Symbolic Name	Channel Number
#1	#1	DX 7	6
#1	#2	Casio	2

Figure 4: Configuration information for a device. Adapted from: Routing MIDI messages in a shared music studio environment (Mosala, 1995).

The MIDINet ID is a unique number between 1 and 10 that allows the MIDINet units to be identified within the system. The port ID is used to identify the port on the MIDI interface (MIDI card) that is to be selected. This ID is in the form of a number from 1 to 4, as the system only caters for MIDINet units to boast 4 MIDI ports. Channel number is used to configure which channel the attached MIDI device is using. As the MIDI specification states that MIDI devices have a maximum of 16 channels, the MIDINet system deals with channel number in the same way. The MIDINet system also allows for a symbolic name to be associated with a MIDINet unit, port and channel combination. The name is to be assigned by the user of the system and is limited to 10 characters.(Mosala, 1995)

MIDINet Unit ID	Port ID	Symbolic Name	Channel Number
#1	#2	Voyetta 1	1
#1	#2	Voyetta 1	2

Figure 5: Configuration Information for a multi-channel transmitter. Adapted from: Routing MIDI messages in a shared music studio environment (Mosala, 1995).

MIDINet Unit ID	Port ID	Symbolic Name	Channel Number
#1	#2	D110 Part 1	1
#1	#2	D110 Part 2	2

Figure 6: Configuration Information for a multi-channel receiver. Adapted from: Routing MIDI messages in a shared music studio environment (Mosala, 1995).

MIDINet Unit ID	Port ID	Symbolic Name	Channel Number
#1	#2	D110 Part 1	1
#1	#3	DX 7	2
#3	#4	Voyetta 3	2

Figure 7: Source and Destination device list. Adapted from: Routing MIDI messages in a shared music studio environment (Mosala, 1995).

M-ID	P-ID	S-Name	C#		M-ID	P-ID	S-Name	C#
#2	#2	Voyetta 1	2	-	#1	#2	Casio	1
#2	#2	Voyetta 2	3	-	#1	#3	DX 110	2

Figure 8: Connection Table. Adapted from: Routing MIDI messages in a shared music studio environment (Mosala, 1995).

When the MIDI interface receives MIDI bytes, the MIDINet system parses these with a MIDI parser, grouping the messages as they are grouped in the MIDI specification:

- MIDI channel messages

    MIDI channel voice messages

    MIDI channel mode messages

- MIDI system messages

    MIDI system common messages

    MIDI system real time messages

    MIDI exclusive messages

It is interesting to note that more than one connection may use the same channel, as MIDINet connections are first identified by MIDINet unit ID, then port, then channel, making it possible to have multiple messages being sent using a particular channel, but communicating with with a number of different MIDI devices. (Mosala, 1995)

3.2 OSC: Open Sound Control

3.2.1 History

Open Sound Control or OSC was developed by Adrian Freed and Mathew Wright as a "protocol for communication among computers, sound synthesizers, and other multimedia devices that is optimized for modern networking technology" (Wright & Freed, 1997).

The OSC protocol was designed independent of its method of transport. The protocol is therefore able to be used on standard network technology such as TCP and UDP, running over Ethernet or 802.11 wireless networks. (Wright, 2005)

3.2.2 Specification

The protocol defines a data transport format that can be used to send control messages over networks. OSC does not define a set of standard messages for devices to use, but rather encourages manufactures and implemented to create and use their own message sets: Schemas. Schemas refer to the configurations that defines the structure of a device. Therefore the address space is application specific. There are no standard schemas at the moment and manufacturers need to define the structure of their device in order to use the OSC messaging standard. (Wright, 2002)

The OSC protocol has been used to simply encapsulate MIDI message for transport over networks, but can be used in many other ways for other uses. (Wright, 2005)

3.2.3 Comparison to MIDI

The designers of OSC state that it is superior to MIDI in several ways. The MIDI standard was never intended for uses over LANs (Local Area Networks), let alone the Internet. OSC provides larger address space for messages, allowing for more complex and specific control messages. Modern synthesizers can sometimes exceed the MIDI specification in terms of channels and parameters, areas where OSC is not limited. (Eales, 2012)

4 Control Applications

4.1 Yamaha Control

Yamaha Pro Audio Inc. developed their own control application, Studio Manager which is currently available in its second version which offers support for a variety of Yamaha products. The Studio Manager software allows users to exert control over compatible Yamaha products that are connect to the controlling computer. Devices such as the Yamaha 02R96, 01V96 and 01X are all compatible with software, allowing a user to adjust parameters, route signals and change levels on these devices remotely. (Yamaha, n.d.)

In the case of the 01V96 and 01X, Yamaha has implemented four layers, allowing these desks to support more mixing and channels than they have physical controls for. This is an issue as engineers would need to switch layers in order to adjust the level on a channel that is not part of the current layer. With the Yamaha control software, viewing more than one layer at a time is not a problem, and thus the computer control provides extra control over the device. (Yamaha, n.d.)

Yamaha have also released an application called StageMix that can provide control for their CL, M7CL and LS9 mixing consoles. The application is designed for the Apple iPad and aims to give the user remote control of the above mentioned mixing consoles. The application requires a wireless connection to an access point connected to the mixing console, but allows the user to move freely within the confines of the wireless network and control the mixing console via the tablet.(Yamaha, n.d.)

4.2 Allen and Heath Control

Allen & Heath Ltd., like Yamaha, provide applications for control of their own brand mixing consoles. Allen & Heath provide both desktop and tablet based applications for their iLive range of digital mixng consoles. iLive Editor is a control application aimed at desktop and laptop computers and is available both for PC and Apple Mac. The computer application communicates with the iLive mixer using a standard Ethernet cable. Wireless control can be achieved by connecting the iLive mixer to wireless access point. Allen & Heath have a version of the application available for the Apple iPad called iLive MixPad, which uses WiFi to connect to mixing consoles attached to a WiFI access point. The iPad application provides touch control for parameters, giving the user a more intuitive and ”closer to the real thing” experience. These applications provide a wide range of control, but are limited to only controlling Allen & Heath iLive Series mixing consoles. (Allen&Heath, n.d.b)

A similar application to the iLiveMix application for Apple iPad, GLD Remote, is available for Allen & Heath's GLD Series of mixing consoles. GLD Remote is aimed at giving total wireless control and allows for a simple iPad to control many GLD mixers. It also allows for multiple iPad to control a single desk, separating control for various sections of the desk to different iPads. (Allen&Heath, n.d.a)

4.3 Matrix Mixer

Philip Foulkes, a student from Rhodes University, developed a concept for a connection management application: The Matrix Mixer. (Foulkes, 2006) The thought behind the concept was that routing by setting sources and destinations on a mixing console could be simplified by viewing the possible connections as a grid.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Main					■	■	■	■	■			■		
Grp 1	■	■	■	■										
Grp 2										■	■			
Grp 3														
Grp 4			■	■									■	■
Aux 1					■	■	■							
Aux 2					■									
Aux 3					■									
Aux 4					■		■							
Aux 5						■								
Aux 6							■	■	■					

Figure 9: Example of matrix based connection management.

In the above figure it is shown that the matrix provides a simple overview of connections, linking inputs (listed along the top in this example) to output (down the side in this example). By selecting an intersection point between an input and an output, a connection can be made. For control of parameters associated with a single input or output, the channel can be selected and a channel parameter page is displayed. Parameters such as level, equalisation settings and panning can be altered from this view. (Foulkes, 2006)

Philip Foulke's Matix Mixer application was originally designed to work with the Yamaha 01X, but its generic design allows for it to be used on a variety of mixing consoles. Mixing consoles are represented using XML (extensible markup language), allowing for the application to take on the form and parameters of any desk. (Foulkes, 2006) This makes the representation of mixing consoles, both old and new, rather simple and making it possible to integrate this application with a wider range of products.

Connection using a propriety format would lead to the application being less generic, and so the MIDI standard was implemented as a method of both synchronising the current state of the application or desk, as well as providing remote control in both directions. The application communicates with the desk using MIDI messages, sent out from the computer using a MIDI interface. The Matrix Mixer application was only developed for Windows computers and no touch implementation is available. (Foulkes, 2006)

4.4 Touch screen devices

For years the graphical user interface (GUI) has been the dominant method of interfacing with computers. (Wigdor & Wixon, 2011) Although GUIs provide a very versatile and workstation-friendly method of providing input, they are seldom intuitive or easy to learn, meaning that users cannot get straight to using applications, but are required to learn the interface, even if only for a while.

With the recent increase in availability of touch screens on general devices such as cellphones, car stereos, tablets, laptops and computers, the need for a new method of interaction was needed. The Natural User Interfaces (NUI) is a concept that has been put forward as a design option for touch screen applications. Everyday users employ their hands to do tasks, and people use their fingers to grab and move small items, it thus seems strange that users need to interact with a mouse in order to control what is on the screen of a computer. Natural user interfaces aim to provide an intuitive look and feel for application allowing users to use an application for the first time, and as the name suggests, have it feel natural. The basis of natural user interfaces is that users need not only click or hold on icons with a touch screen, but slide or swipe over the screen in a much freer way. Many multitouch devices provide the ability to understand gestures, giving the touch screen the ability to be interacted with on a gestural level.

Since the development of touch screens, one typical use for the interface has been that of replacing real world control surfaces with on-screen applications. Large arrays of buttons and gauges that may appear on a machine's control panel are now capable of being placed within an application. This has brought forth further option for improvement, such as the ability to have different views, or expand various components to view finer controls or details.

4.4.1 Apple iPad

In 2007, Apple Inc. released the iPhone, a touch screen mobile phone. The phone gained popularity quickly as users were able to use the touch screen and move more freely within the operating system (OS). Apple had originally intended to release a tablet computer, but had instead decreased its size to make a play for the mobile phone market. (Fling, 2009)

In 2010 the first iPad was released by Apple Inc. The device has a 9.7" multitouch display, Bluetooth and 802.11 wireless networking (WiFi). This combination of features made the iPad a perfect platform for mobile application, particularly applications that integrated with a larger system. Many developers took to creating iPad applications, but the music production industry in particular accepted the iPad and pushed for more audio application development. Music oriented application for both creating sounds (synthesis) as well as arranging and producing became popular and some manufacturers even started to provide propriety control software for their systems. (EUMLab, n.d.)

5 Conclusion

This literature review has discussed topics including communications protocols, control applications and the progression of natural user interfaces. These topics contribute to the understanding of the requirements for a system for controlling mixing consoles.The system would be generic enough to not limit users to specific devices, but that is powerful enough allow for fully functional remote control.

This literature review has shown how existing technologies such as MIDI is currently capable of providing control for mixing consoles, as well as being used widely for synthesiser control. It also shows how current control applications, while providing an excellent level of control, are proprietary systems and lack the generic nature required for a complete system of control. The discussion on mobile touch screen computing shows a more natural interface is not only possible, but will result in decreased learning time.

All of these topics provide valuable insight when developing a comprehensive, generic and intuitive system of control.

References

Allen&Heath. *GLD Remote.* http://www.allen-heath.com/UK/Products/Pages/ProductDetails.aspx?catId=GLDSeries&ProductId=GLDRemote&Sub

Allen&Heath. *iLive MixPad.* http://www.allen-heath.com/UK/Products/Pages/ProductDetails.aspx?CatId=iLiveSeries&ProductId=iLiveMixPad&Su

DeFuria, Steve, & Scacciaferro, Joe. 1990. *MIDI Programmer's Handbook.* Foster City, CA, USA: IDG Books Worldwide, Inc.

Eales, Andrew; Foss, Richard. 2012 (10). Service Discovery Using Open Sound Control. *In: Audio Engineering Society Convention 133.*

EUMLab. *DAW Remote HD: Control your DAW wirelessly.* http://eumlab.com/daw-remote-hd/.

Fling, B. 2009. *Mobile Design and Development: Practical concepts and techniques for creating mobile sites and web apps.* O'Reilly Media.

Foulkes, Philip James. 2006. *A Grid Patch-Bay for Audio Mixers.*

Hosken, D. 2010. *An Introduction to Music Technology.* Taylor & Francis.

Huber, David Miles. 2012. *The MIDI manual: a practical guide to MIDI in the project studio.* Focal Press.

Jacobs, Gabriel, & Georghiades, Panicos. 1991. *Music and new technology: the MIDI connection.* Sigma Pr.

Kirk, Ross, & Hunt, Andy. 1999. *Digital sound processing for music and multimedia.* Focal Pr.

Mosala, Thabo Jerry. 1995. *Routing MIDI messages in a shared music studio environment.*

Rothstein, Joseph. 1995. *MIDI: A comprehensive introduction.* Vol. 7. AR Editions, Inc.

Wigdor, D., & Wixon, D. 2011. *Brave NUI World: Designing Natural User Interfaces for Touch and Gesture.* Elsevier Science.

Wright, Matthew. 2002. Open Sound Control 1.0 Specification.

Wright, Matthew. 2005. Open Sound Control: an enabling technology for musical networking. *Organised Sound*, **10**(2005/12/01), 193–200.

Wright, Matthew, & Freed, Adrian. 1997. Open Sound Control: A New Protocol for Communicating with Sound Synthesizers. *Pages 101–104 of: International Computer Music Conference.* Thessaloniki, Hellas: International Computer Music Association.

Yamaha. *LS9 Stage Mix User Guide.* http://download.yamaha.com/file/55009.