VB MIDI PIANO
By Arthur Edstrom
Artic Software
September 3, 1993

Contact me at:
Artic Software
PO Box 28
Waterford, WI 53185-0028 USA

Compuserve 75410,2203
or EXEC-PC

Hello Visual Basic Programmer ,
VB MIDI PIANO is a sample Visual Basic program that shows how to begin to access some of the low level midi capibilities of Windows using Visual Basic. I have included an outline for a complete article on this subject if a publication is interested in having me developing this further. This version of VB MIDI PIANO doesn't have all the features that are in the outline, but it gets pretty close. In this sample I'm also only sending Short MIDI Messages, but Long Messages are not too hard. You will also see that I'm sending everything through the MIDI Mapper, but this can be modified.

Note: The PATCH.INI file needs to be in your WINDOWS directory. I did not put in the path handling routines yet. Let me now if this sample is of any help.

Arthur Edstrom
Artic Software

Create a Playable Graphic MIDI Piano Keyboard with Visual Basic

Synopsis

An article will be written that explains how to use Visual Basic to create a playable graphic MIDI piano keyboard. This program will work with any sound card that supports MIDI or has an on-board synthesizer. The Creative Lab Sound Blaster, Microsoft Sound System, Turtle Beach Multisound and dozens of other sound cards are in this catagory. There are currently millions of sound card users, with over 200,000 sound cards being sold per month. With BASIC being a language that hobbists have used for years, promoting Visual Basic for use by hobbists for controlling their sound cards seems like a great idea. This article will be unique in that we will make "low level" API calls to the multimedia extension in Windows 3.1 instead of just using the MCI. The MCI does not allow for control of individual MIDI events.

The article will start with a basic introduction on MIDI (Musical Instrument Digital Interface) to explain what MIDI is. This will be a brief explanation to cover just enough MIDI information to get the readers going on the programming project. An explanation of MIDI events and a table showing MIDI events that are used in this project should be shown.

With an understanding of MIDI, we can then address using Visual Basic to access the sound card installed in your computer. The correct way to approach communicating with your sound card, is to use the midiOutGetNumDevs and midiOutGetDevCaps to find out if a MIDI device is installed. If a MIDI device is installed, what are the capabilities of this device? We will also look at a short-cut which involves using just the MIDI Mapper to access the MIDI device. This portion of the article will conclude with the creation of a simple program that allows you to open a MIDI device, send a MIDI note and then close the MIDI device.

The fun part of this project begins when you start to turn your computer into a true MIDI musical instrument. We will use a simple panel control (THREED.VBX) to draw a nice graphic

piano display. (We could use many of the different controls that are available in Visual Basic, but the panel control looks great.) We will setup controls for MIDI Channel, Volume, Panning, Octave control of the Graphic Piano and Patch selection. The patch selection will access a PATCH.INI file containing 128 General MIDI patch names.   Once we have completed this project the user will be able to play any musical note on any MIDI channel just by dragging and dropping the mouse on the Graphic Piano display. Full control of Volume, Panning, Octave control and Patch selection for each MIDI channel will be available.

It is amazing that with Visual Basic and few API calls, we can create MIDI music. There is no need to develop a DLL. We are no longer forced to develop MIDI applications only in C. Let your sound card and VB bring music to your ears. Create a Playable Graphic MIDI Piano Keyboard with Visual Basic.

Outline
I.   Brief overview of MIDI. (Musical Instrument Digital Interface)
        A.   What is MIDI?
        B.   Explain what MIDI events are. (display a table of all MIDI events)
                 1.   Note on
                          a.   Channel number
                          b.   Note value
                          c.   Volume value
                 2.   Patch
                          a.   Channel number
                          b.   Patch number
                 3.   Pan
                          a.   Channel number
                          b.   Pan value

II.   Develop routines for MIDI output.
        A.   Detect and display capabilities of MIDI devices installed in computer.
                 1.   Get the number of devices in system using midiOutGetNumDevs.
                 2.   Get the capabilities of each device using midiOutGetDevCaps.
                 3.   Simpler option is to just use the MIDI Mapper and assume MIDI
                          device is installed.
        B.   Select and open one of the MIDI Outputs installed in computer.
        C.   Set up and play a note from a command button.

III.   Develop routines for graphic MIDI piano keyboard.
        A.   Draw a piano keyboard.
        B.   Turn piano into an active instrument.
        C.   Make keys on keyboard change color when played.
                 1.   Briefly explain modulo arithmetic.
                 2.   Global color constants.
        D.   Control MIDI Channel.
                 1.   Use vertical scroll bar
                 2.   Set value 1 - 16
                 3.   Set default value to 1
        E.   Control volume
                 1.   Use vertical scroll bar
                 2.   Set value 0 - 127
                 3.   Set default value to 80
        F.   Control octave for graphic piano
                 1.   Use vertical scroll bar
                 2.   Set value 0 - 4
                 3.   Set default value to 0
        G.   Control patch changes
                 1.   Use MIDI values 0 - 127

2. Explore General MIDI Patches
   a. Create a PATCH.INI file
   b. Access PATCH.INI for each patch change
H. Control panning
   1. Use horizonal scroll bar
   2. Set value 0 - 127
   3. Set default value to 64
I. Options
   1. Drag and drop monophonic
   2. Drag and drop polyphonic