Notes and Volts
Electronics, Guitars, and Geekery

Home | LIVE | Amazon Store | TEKBRANE | Steal My Music | About | Contact | Disclosure

Saturday, 28 January 2012

Fun with Arduino - Midi Input Basics

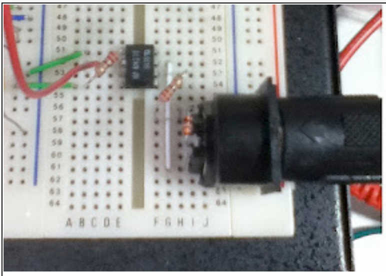
Midi In - "Hello World!"

**** This article has been updated! Check out the NEW IMPROVED VERSION. ****

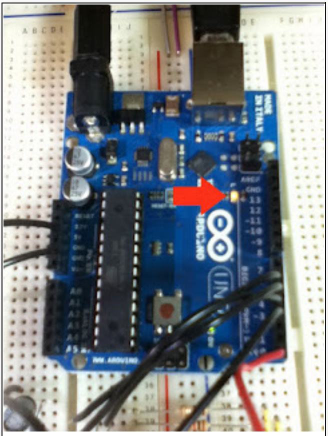
Check out this video for a detailed Analysis of the Midi Circuit.

For a real world look at this circuit in action, check out the NaV-1 Arduino synth construction series.

Have you ever been working on a Arduino project and suddenly thought 'This thing could really use a MIDI Input'! This exact thing just happened to me. Not wanting to re-invent the wheel, I began searching around for a Library that would help. I came across the Arduino Midi Library which seemed to fit the bill. It took me a while to actually get the thing going, so I thought I would write a quick post outlining the steps to get a simple test circuit working. The following program and circuit will simply Flash the LED connected to Pin 13 on the Arduino Board when you press a note on a Midi Keyboard. But, that's really all you really need to confirm that you are correctly receiving Midi commands with your Arduino.



The Midi In Circuit



Flashing LED means you Win!!

The Arduino Midi Library

Follow Notes and Volts

Follow on Facebook

Follow on Google+

Follow on twitter

Follow on Pinterest

Subscribe via RSS

Labels

Electronics Music MIDI Geekery CNC Guitar

Search

Search

Blog Archive

January 2012 (4)

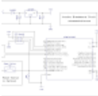
Popular Posts

Build the Auduino Granular Synth - Part 1

Parts, Prepping and Drilling In this tutorial we will take a look at building a stand-alone version of the Auduino Granular Synthesizer ...

Fun with Arduino - Midi Input Basics

Midi In - "Hello World!" ** This article has been updated! Check out the NEW IMPROVED VERSION . ** Check out this video for a ...



Fun with Arduino - Arduino on a Breadboard

How to remove the 'Arduino' from your Arduino project In this tutorial, we are going to build a 'stand-alone' circuit tha...

Fun with Arduino - The Arduino as ISP

Time to BURN! In the previous tutorial, we learned how to build a 'stand-alone' Arduino circuit on a breadboard using an ATMEGA3...

Midi for the Arduino - Build a Midi Input Circuit

Less talking, more building! In this installment we are going to assemble a Midi Input Circuit and connect it to your Arduino UNO.

First go to this Link and download the Arduino Midi Library files.

Unzip the downloaded folder. There are two folders inside. For Windows, copy the folder called "MIDI" and paste it into your Arduino "libraries" folder. Mine was located inside the "arduino-0022" folder. Quit and restart your Arduino IDE program. Go to the Menu and open Sketch > Import Library. You should see "MIDI" as one of the choices.

Copy and paste the code at the end of this article into a new Sketch. The code is commented, so give it a quick read through.

Here are some of the key commands:

MIDI.begin(MIDI_CHANNEL_OMNI);
This initializes the Midi Library. The MIDI_CHANNEL_OMNI parameter sets the library to listen to all Midi Channels. MIDI.begin(2) would set it to listen to Channel 2 only.

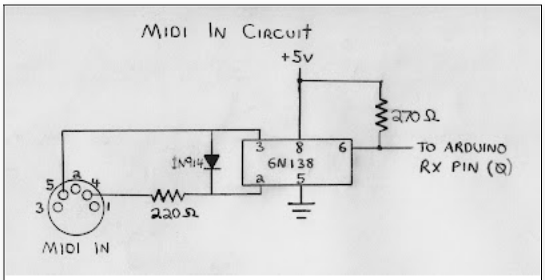
MDI.setHandleNoteOn(MyHandleNoteOn);
This is an import command! The Arduino Midi Library uses something called 'Callbacks'. When a Midi event occurs, the Library will Call a function to handle it. This command tells the Library to call the 'MyHandleNoteOn' function when a **'Note On'** Midi event is detected. There are many callback functions in the Library to handle the many types of Midi events (Clock, Pitch Bend, Program Change, Etc..). Use the MIDI.set... command to point to the functions you require.

void MyHandleNoteOn(byte channel, byte pitch, byte velocity)
This is the function I created to be called when a Midi **Note On** event is detected. This is the 'meat' of your program. In this test program, I just have it flash the LED on the Arduino board. But you could just as easily have it play a note on your home made Synth circuit, Flash a spotlight on your Midi controlled lighting rig, or even command your Midi controlled Robotic Gorilla to enter 'Rampage' Mode. The sky is the limit.

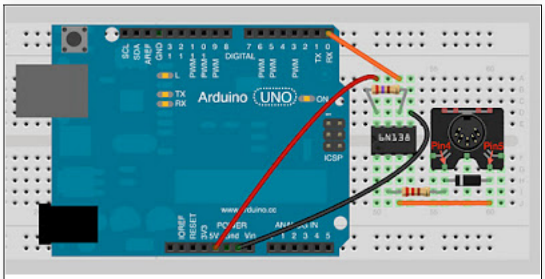
MIDI.read();
This is the only function in the main loop of the program. It just checks the input buffer for any received Midi commands and passes them to the correct function.

The Hardware

The MIDI standard spells out the circuit that should be used for a MIDI INPUT so lets look at that first.



Arduino Midi In Circuit Schematic



Midi In Circuit on the Breadboard

This very simple circuit uses a 6N138 optocoupler chip. This device basically electrically isolates your circuit from the incoming Midi signal. The 1N914

The Ci...

Arduino MIDI Controller: Potentiometers
Take Control In this installment of the MIDI for the Arduino series, we will apply the knowledge we have gained from previous chapters a...

Arduino MIDI Controller: Multiplexers
More, More, More!! Over the last couple of tutorials we have been building up a MIDI Controller using a trusty Arduino UNO board. So far,...

Build the Auduino Granular Synth - MIDI Upgrade
Ever since I posted the original Auduino Granular Synth tutorial, I have been receiving requests to modify the circuit for MIDI control. ...

Arduino MIDI Controller: Buttons
Push It! In this installment of the MIDI for the Arduino Series, we will add Push Buttons to the Midi Controller we started building in th...

Build the Auduino Granular Synth - Part 2
Panel Art, Wiring, and Finishing Touches In Part 1 of this project we looked at building and programming the circuit board, drilling the...

diode protects the chip from an incorrectly wired Midi cable. Plug the output of the optocoupler (Pin 6) into the RX (Pin 0) socket on your arduino board. **Note: Be sure to correctly identify pin 4 and 5 on the Midi In Jack.**

Also note that the RX/TX pins and the USB Port on the Arduino Board share the same signals. So, you will need temporarily remove the wire from the RX Pin 0 on the Arduino Board to upload a program. Then remove the USB Cable from the computer and replace the RX wire when you run the program with a Midi Input.

That's really all there is to it! Connect the Midi Out port on a Midi Keyboard to the Midi In Jack on your Arduino. If the little LED next to Pin 13 on the Arduino Board flashes when you hit a key, you are correctly recieving the Midi Data. Have fun!

For a real world look at this circuit in action, check out the NaV-1 Arduino synth construction series.

Copy the following code and paste into a new Arduino Sketch:

```
#include <MIDI.h>  // Add Midi Library

#define LED 13    // Arduino Board LED is on Pin 13

// Below is my function that will be called by the Midi Library
// when a MIDI NOTE ON message is received.
// It will be passed bytes for Channel, Pitch, and Velocity
void MyHandleNoteOn(byte channel, byte pitch, byte velocity) {
  digitalWrite(LED,HIGH);  //Turn LED on
  if (velocity == 0) {//A NOTE ON message with a velocity = Zero is actualy a NOTE OFF
    digitalWrite(LED,LOW);//Turn LED off
  }
}

void setup() {
  pinMode (LED, OUTPUT); // Set Arduino board pin 13 to output
  MIDI.begin(MIDI_CHANNEL_OMNI); // Initialize the Midi Library.
// OMNI sets it to listen to all channels.. MIDI.begin(2) would set it
// to respond to channel 2 notes only.
  MIDI.setHandleNoteOn(MyHandleNoteOn); // This is important!! This command
  // tells the Midi Library which function I want called when a Note ON command
  // is received. in this case it's "MyHandleNoteOn".
}

void loop() { // Main loop
  MIDI.read(); // Continually check what Midi Commands have been received.
}
```

Posted by Dave at 1:27 am

Labels: Electronics, Music

209 comments:

--((()))--- 1 April 2012 at 03:37

excellent!

Reply

Replies

Dave 2 April 2012 at 01:18

Thanks! Glad you enjoyed it.

Anonymous 12 April 2012 at 18:52

Your breadboard diagram has an error. The diode should be at the optocoupler input not at the connector.