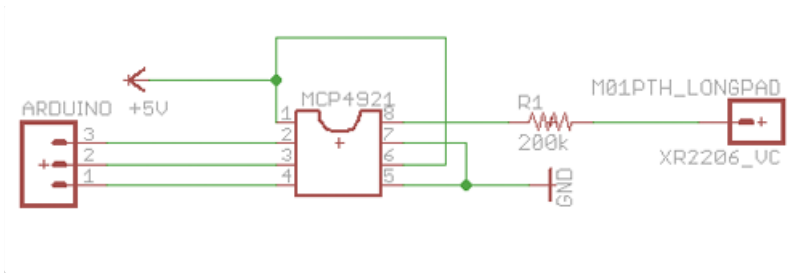


MCP4921 12bit DAC voltage controller

The MCP4921 controls the sweep voltage for the XR2206. The sweep voltage and the variable resistor together are controlling the output frequency. Precise musical output frequency should be feasible this way. I will be using [this table](#) to check if the frequencies are correct.



The MCP4921 connects to Arduino pins 10, 11 and 13.

MCP pin 2 > Arduino 10

MCP pin 3 > Arduino 13

MCP pin 4 > Arduino 11

Derived from [mrBooks instructions!](#)

```
#define DATAOUT    11  //MOSI
#define SPICLOCK    13  //sck
#define SLAVESELECT 10  //ss
#define ledPin      2

int interval;
int incomingByte = 0;

void setup() {
  // -----
  pinMode(DATAOUT,    OUTPUT);
  pinMode(SPICLOCK,   OUTPUT);
  pinMode(SLAVESELECT, OUTPUT);
  pinMode(ledPin,     OUTPUT);

  sendIntValueSPI(1000);

  // led on
  digitalWrite(ledPin, HIGH);

  // start serial interface
  Serial.begin(9600);
}

void sendIntValueSPI(int value) {
  // -----

  // initiate data transfer with 4921
  digitalWrite(SLAVESELECT, LOW);

  // send 4 bit header
  sendSPIHeader();

  // send data
  for(int i=11; i>=0; i--){
    digitalWrite(DATAOUT, ((value & (1<<i))) >> i);
    sendSPIClock();
  }
}
```

```

// finish data transfer
digitalWrite(SLAVESELECT,HIGH);
}

void sendSPIHeader() {
// -----
// bit 15
// 0 write to DAC *
// 1 ignore command
digitalWrite(DATAOUT,LOW);
sendSPIClock();
// bit 14 Vref input buffer control
// 0 unbuffered *
// 1 buffered
digitalWrite(DATAOUT,LOW);
sendSPIClock();
// bit 13 Output Gain selection
// 0 2x
// 1 1x *
digitalWrite(DATAOUT,HIGH);
sendSPIClock();
// bit 12 Output shutdown control bit
// 0 Shutdown the device
// 1 Active mode operation *
digitalWrite(DATAOUT,HIGH);
sendSPIClock();
}

void sendSPIClock() {
// -----
digitalWrite(SPICLOCK,HIGH);
digitalWrite(SPICLOCK,LOW);
}

void loop() {
// -----
if(Serial.available() > 0)
{
interval = GetStringNumber();
Serial.print("New voltage: ");
Serial.println(interval,DEC);
sendIntValueSPI(interval);
}

}

int GetStringNumber()
{
int value = 0;

while(1)
{
/*Read a byte as it comes into the serial buffer*/
char byteBuffer = Serial.read();

if(byteBuffer > -1) //Is the data a valid character?
{
if(byteBuffer >= '0' && byteBuffer <= '9') //Is the character a digit?
/*Yes, shift left 1 place (in decimal), and add integer value of character (ASCII value - 48)*/
value = (value * 10) + (byteBuffer - '0');
else
/*No, stop*/
break;
}
}
return value;
}

```

This entry was posted in [arduino](#), [newsynth](#) on 5 June 2010.

[← LCD prototype](#)

[MCP42100 Potmeter with SPI →](#)

One thought on “MCP4921 12bit DAC voltage controller”

there is a problem with this line in your code !

```
if(Serial.available() > 0).....
```

and : `int incomingByte = 0;` is not used in your code

Comments are closed.

CATEGORIES

[arduino](#) (28)
[edusynth](#) (1)
[juno106](#) (14)
[newsynth](#) (28)
[research](#) (1)
[sine](#) (2)
[ubuntu](#) (4)
[Uncategorized](#) (13)

BLOGROLL

[github](#)
[hackernews](#)
[matrixsynth](#)

[Proudly powered by WordPress](#)