// include SPI, MP3 and SD libraries

#include <SPI.h>

#include <Adafruit\_VS1053.h>

#include <SD.h>

#include <SoftwareSerial.h>

SoftwareSerial mySerial(5, 2); // RX, TX

// define the pins used

//#define CLK 13 // SPI Clock, shared with SD card

//#define MISO 12 // Input data, from VS1053/SD card

//#define MOSI 11 // Output data, to VS1053/SD card

// Connect CLK, MISO and MOSI to hardware SPI pins.

// See http://arduino.cc/en/Reference/SPI "Connections"

// These are the pins used for the breakout example

#define BREAKOUT\_RESET 9 // VS1053 reset pin (output)

#define BREAKOUT\_CS 10 // VS1053 chip select pin (output)

#define BREAKOUT\_DCS 8 // VS1053 Data/command select pin (output)

// These are the pins used for the music maker shield

#define SHIELD\_RESET -1 // VS1053 reset pin (unused!)

#define SHIELD\_CS 7 // VS1053 chip select pin (output)

#define SHIELD\_DCS 6 // VS1053 Data/command select pin (output)

// These are common pins between breakout and shield

#define CARDCS 4 // Card chip select pin

// DREQ should be an Int pin, see http://arduino.cc/en/Reference/attachInterrupt

#define DREQ 3 // VS1053 Data request, ideally an Interrupt pin

Adafruit\_VS1053\_FilePlayer musicPlayer =

// create breakout-example object!

// Adafruit\_VS1053\_FilePlayer(BREAKOUT\_RESET, BREAKOUT\_CS, BREAKOUT\_DCS, DREQ, CARDCS);

// create shield-example object!

Adafruit\_VS1053\_FilePlayer(SHIELD\_RESET, SHIELD\_CS, SHIELD\_DCS, DREQ, CARDCS);

void setup() {

Serial.begin(9600);

Serial.println("Adafruit VS1053 Simple Test");

if (! musicPlayer.begin()) { // initialise the music player

Serial.println(F("Couldn't find VS1053, do you have the right pins defined?"));

while (1);

}

Serial.println(F("VS1053 found"));

if (!SD.begin(CARDCS)) {

Serial.println(F("SD failed, or not present"));

while (1); // don't do anything more

}

// list files

printDirectory(SD.open("/"), 0);

// Set volume for left, right channels. lower numbers == louder volume!

musicPlayer.setVolume(50,50);

// Timer interrupts are not suggested, better to use DREQ interrupt!

//musicPlayer.useInterrupt(VS1053\_FILEPLAYER\_TIMER0\_INT); // timer int

// set the data rate for the SoftwareSerial port

mySerial.begin(9600);

#if 0

// If DREQ is on an interrupt pin (on uno, #2 or #3) we can do background

// audio playing

musicPlayer.useInterrupt(VS1053\_FILEPLAYER\_PIN\_INT); // DREQ int

// Play one file, don't return until complete

Serial.println(F("Playing track 001"));

musicPlayer.playFullFile("/M1.mp3");

// Play another file in the background, REQUIRES interrupts!

Serial.println(F("Playing track 002"));

musicPlayer.startPlayingFile("/M2.mp3");

#endif

}

void loop() {

while (mySerial.available()) {

int tag=mySerial.read();

Serial.println(tag);

switch(tag){

case 1 : musicPlayer.playFullFile("/M1.mp3"); break;

case 2 : musicPlayer.playFullFile("/M3.mp3"); break;

}

}

}

/// File listing helper

void printDirectory(File dir, int numTabs) {

while(true) {

File entry = dir.openNextFile();

if (! entry) {

// no more files

//Serial.println("\*\*nomorefiles\*\*");

break;

}

for (uint8\_t i=0; i<numTabs; i++) {

Serial.print('\t');

}

Serial.print(entry.name());

if (entry.isDirectory()) {

Serial.println("/");

printDirectory(entry, numTabs+1);

} else {

// files have sizes, directories do not

Serial.print("\t\t");

Serial.println(entry.size(), DEC);

}

entry.close();

}

}

<br style="clear: both;" />