

Textul și imaginile din acest document sunt licențiate

Attribution-NonCommercial-NoDerivs

CC BY-NC-ND



Codul sursă din acest document este licențiat

Public-Domain

Ești liber să distribui acest document prin orice mijloace consideri (email, publicare pe website / blog, tipărire, sau orice alt mijloc), atât timp cât nu aduci nici un fel de modificări acestuia. Codul sursă din acest document poate fi utilizat în orice fel de scop, de natură comercială sau nu, fără nici un fel de limitări dar autorii nu își asumă nici o răspundere pentru pagubele pricinuite de implementările realizate de utilizatori. Schemele și codul sursă au un rol educativ și nu sunt gândite pentru a fi utilizate în mediu de producție (industrial, casnic sau comercial).

Utilizarea bibliotecii LUFA

LUFA (Lightweight USB Framework for AVRs) [1] este o bibliotecă ce implementează stiva USB pentru microcontrolerele Atmel AVR ce dețin un port hardware USB (AVR USB Controller) permițând implementarea de dispozitive USB variate: Android Accessory Host, Audio In Device, MIDI Device, Generic HID Device, Joystick Device, Keyboard Device, Printer Host, Virtual Serial Device etc. Biblioteca LUFA este componenta software care stă la baza funcționării majorității plăcilor Arduino: în cazul Arduino Uno / Arduino Mega biblioteca este utilizată de firmware-ul ce permite microcontrolerelor ATmega8U2/16U2 să realizeze comunicația USB-to-serial și să încarce programul în memoria microcontrolerului central; în cazul Arduino Leonardo / Arduino Micro, plăci echipate cu un microcontroler ATmega32U4, biblioteca este utilizată direct de bootloader-ul plăcii permițând atât comunicația USB-to-serial cât și încărcarea programului.

Plăcile de dezvoltare Arduino nu sunt singurele aplicații embedded ce utilizează biblioteca LUFA, mai jos sunt trecute în revistă câteva dintre multele proiecte bazate pe această bibliotecă:

The Smart Card Detective [2] este un sistem (imagine alăturată) ce permite interceptarea, înregistrarea și modificarea comunicației dintre un smartcard și un cititor de carduri (inclusiv de tip EMV).



RFM12B USB light – Stick [3] un dispozitiv de comunicație pentru PC-uri bazat pe microcontrolerul ATmega32U4 și modulul radio FSK RFM12B.

The Finch [4] este un robot educațional gândit pentru a fi folosit în explicarea timpurie a noțiunilor de programare, robotică, sisteme de automatizare.

AD9833-based USB Function Generator [5] este un sistem programabil ce permite generarea unor semnale analogice.

FlySight [6] (imagine alăturată) este un sistem ce permite înregistrarea pe un card de memorie a coordonatelor GPS corelate cu viteza și accelerația pe 3 axe.



În cadrul materialului de față vă propunem detalierea a două sisteme bazate pe biblioteca LUFA: (1) transformarea unei plăci Arduino Uno într-un dispozitiv MIDI prin reprogramarea microcontrolerului 16U2 și (2) un cititor de carduri bazat pe microcontrolerul ATmega32U4.

Realizarea unui dispozitiv de tip MIDI

Dispozitivele MIDI [7], [8] sunt dispozitive digitale capabile să genereze sunete sau să transmită comenzi de generare a acelor sunete către alte dispozitive digitale (putem vorbi de un sintetizator sau de o claviatură conectată la un sistem de calcul ce rulează un software de sinteză a notelor). În cadrul proiectului nostru vom transforma o placă de dezvoltare Arduino Uno într-un dispozitiv ce se va conecta la PC prin intermediul unei conexiuni USB și va transmite comenzi specifice MIDI [9] către un software specializat. Pentru a putea modifica comportamentul USB al plăcii (tipul de periferic USB văzut de sistemul de operare) este necesar să modificăm firmware-ul microcontrolerului 16U2 (prezent pe Uno R3, versiunile anterioare se bazează pe microcontrolerul 8U2). Pentru a reprograma microcontrolerul 16U2 există două posibilități:

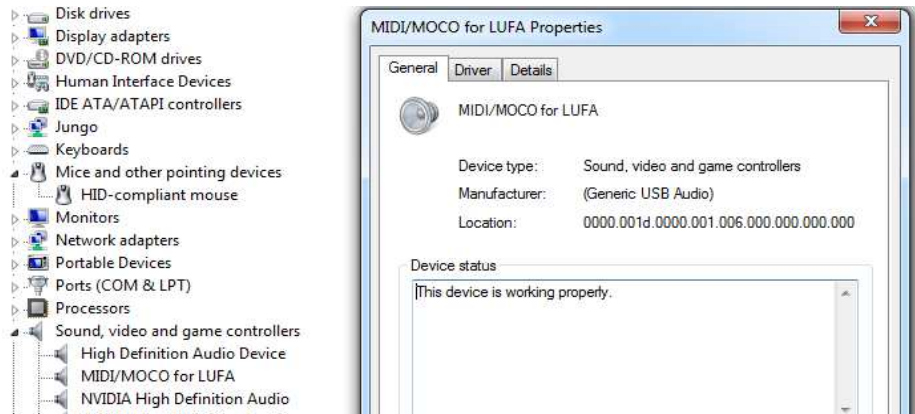
- prin intermediul conexiunii USB utilizând un software de programare DFU [10];
- utilizând portul ICSP [11] al microcontrolerului 16U2/8U2 (imaginea de mai jos surprinde poziția portului ICSP la diverse variante de plăci Arduino Uno) utilizând un programator ISP (USBtiny [12] – modalitate utilizată în explicațiile următoare – sau USBasp [13]) sau o altă placă Arduino Uno pe post de programator ISP.



Firmware-ul microcontrolerului 16U2/8U2 va fi rescris cu Moco for LUFA [14], software ce va transforma placa Arduino Uno într-un dispozitiv MIDI – se va download-a arhiva MocoLUFA-110123.tgz și se va utiliza binarul MIDI.hex din directorul HEX. Binarul este compilat pentru microcontrolerul 8U2 dar poate fi utilizat și pentru 16U2 (va funcționa pe orice variantă de placă Arduino Uno). Utilizând utilitarul avrdude [15] se va încărca binarul utilizând comanda (programator USBtinyISP, placă de dezvoltare Arduino Uno R3 echipată cu varianta 16U2):

```
avrdude -c usbtiny -p m16u2 -U flash:w:MIDI.hex
```

După încărcare noului firmware, la conectarea plăcii sistemul de operare o va identifica ca dispozitiv MIDI (captură de ecran pe pagina următoare) și va instala driverul implicit (nu este necesar un driver special).



Atenție!!! După modificarea firmware-urului microcontrolerului 16U2/8U2 placa Arduino Uno va funcționa doar ca dispozitiv MIDI, nu va mai fi posibilă utilizarea ca o placă de dezvoltare obișnuită – nu va mai fi posibilă reprogramarea acesteia. Pentru revenirea la funcționalitatea inițială se va reîncărca firmware-ul original al plăcii pe microcontrolerul 16U2/8U2 utilizând comanda (firmware-ul original se găsește în directorul de instalare al mediului Arduino IDE în subdirectorul hardware\arduino\avr\firmware\atmegaxxu2):

```
avrdude -c usbtiny -p m16u2 \  
-U flash:w:Arduino-COMBINED-dfu-usbserial-atmega16u2-Uno-Rev3.hex
```

Din aceste motiv încărcarea programului pe placa Arduino trebuie făcută înainte de modificarea firmware-ului pentru microcontrolerul 16U2/8U2. Programul utilizat va trimite pe serială diverse note conform protocolului MIDI [7] – exemplu preluat din [8]:

```
void setup() {  
    Serial.begin(31250);  
}
```

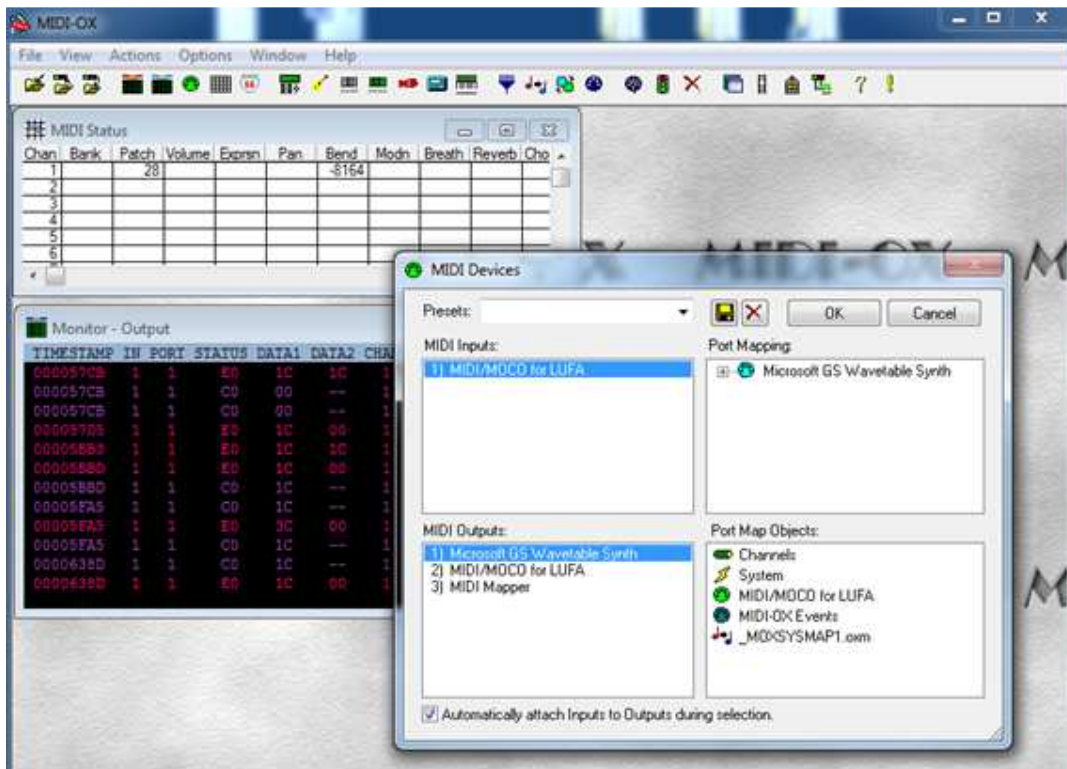
```

void loop() {
    for (int note = 0x1E; note < 0x5A; note ++) {
        noteOn(0x90, note, 0x45);
        delay(100);
        noteOn(0x90, note, 0x00);
        delay(100); }
}

void noteOn(int cmd, int pitch, int velocity) {
    Serial.write(cmd);
    Serial.write(pitch);
    Serial.write(velocity);}

```

Pentru a verifica funcționarea dispozitivului MIDI se poate utiliza orice software specializat, de exemplu: MIDI-OX [16] – utilitar MIDI gratuit. Captura de ecran următoare prezintă modul de configurare a dispozitivului MIDI și captura comenzilor emise de dispozitivul MIDI Arduino utilizând MIDI-OX.



Plăci de dezvoltare bazate pe microcontrolerul ATmega32U4

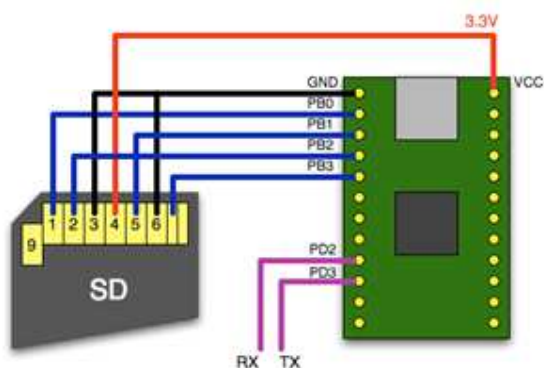
Următoarea generație de plăci de dezvoltare Arduino înlocuiește combinația de microcontroler AVR (ATmega328P – Arduino Uno/Ethernet/Mini/Nano sau ATmega2560 – Arduino Mega) și microcontroler AVR cu suport USB (ATmega16U2 sau ATmega8U2), utilizat pentru conversia USB-to-serial necesară programării, cu un singur microcontroler ce include atât performanțele necesare funcționării plăcii de dezvoltare cât și conectivitatea necesară conexiunii USB. Noua generație de plăci de dezvoltare Arduino utilizează microcontrolerul ATmega32U4 [17] pentru realizarea ambelor sarcini, un exemplu de astfel de placă de dezvoltare este Arduino Leonardo [18] (imagine alăturată) – placă de dezvoltare cu formă și performanțe similare cu placa Arduino Uno dar care este echipată cu un singur microcontroler ATmega32U4 ce rulează atât bootloader-ul de încărcare USB cât și programul scris de utilizator. La fel ca și în cazul microcontrolerului ATmega16U2/8U2 (ce echează Arduino Uno) firmware-ul ce rulează pe microcontrolerul ATmega32U4 se bazează pe biblioteca LUFA. Din acest motiv placa de dezvoltare Arduino Leonardo permite utilizatorului, spre deosebire de placa Arduino Uno, să scrie programe utilizând mediul Arduino IDE care să transforme placa de dezvoltare într-un dispozitiv de tip tastatură sau mouse [19] fără a suprascrie firmware-ul original al plăcii (facilitatea de a se comporta ca un dispozitiv HID USB este deja inclusă în firmware).



Placa Arduino Leonardo nu este singura placă din familia Arduino care este echipată cu microcontrolerul ATmega32U4, alte exemple de astfel de plăci sunt: Arduino Micro, Arduino Yun, Arduino Robot, Arduino Esplora. O placă de dezvoltare ce trebuie de asemenea menționată este Teensy 2.0 [20] (imagine alăturată) deoarece este una dintre cele mai mici plăci de dezvoltare bazate pe ATmega32U4 fiind alegerea cea mai bună pentru proiectele ce au constrângeri legate de dimensiune și din care placa de dezvoltare nu se mai recuperează. Un alt avantaj major al acestei plăci de dezvoltare este dat de componentele software suplimentare față de o placă de dezvoltare Arduino Leonardo dar compatibile cu mediul Arduino IDE [22]. Pe baza acestei plăci vom prezenta dispozitivul următor.



Realizarea unui dispozitiv de tip Cititor de Carduri



Dispozitivul prezentat este bazat pe exemplul [21]. Plecând de la exemplul Mass Storage conținut în biblioteca LUFA, sistemul implementează un dispozitiv USB de citire a unui card de memorie SD (un card reader) combinând exemplul cu biblioteca de lucru cu un card de memorie. Schema de interconectare a componentelor este prezentată în imaginea alăturată. Pentru compilarea / recompilarea proiectului este

necesară instalarea mediului WinAVR și se realizează prin comanda make all în consolă de comandă în directorul LowLevelMassStorage+SD al proiectului. Fișierul MassStorage.hex rezultat în urma compilării proiectului se încarcă în memoria microcontrolerului ATmega32U4 utilizând un programator ICSP (inclusiv o placă Arduino Uno pe post de programator ISP). După încărcarea noului firmware, placa de dezvoltare va fi văzută de sistemul de operare ca un dispozitiv de stocare și se vor putea efectua operațiile de bază cu cardul de memorie (citire, scriere, formatare).

Un proiect similar (o placă Teensy 2.0 programată să se comporte ca un dispozitiv de stocare) se găsește la adresa [23]. Acest proiect utilizează exact același program ca și exemplul precedent dar explică în mod detaliat modul de conectare a unui adaptor SD (Teensy SD adaptor) și modul în care tot ansamblul de componente poate fi integrat într-o carcasă USB astfel încât să rezulte o formă finală elegantă (imagine alăturată). O idee interesantă lansată de acest proiect este posibilitatea de a realiza dispozitive mixte (HID+MassStorage) ce pot fi utilizate în lansarea unor atacuri informatice (un mouse sau o tastatură ce conține un virus în memoria internă – un troian hardware).



Referințe on-line

[1] LUFA

<http://www.fourwalledcubicle.com/LUFA.php>

[2] The Smart Card Detective

<https://github.com/rob6969/smartcarddetective>

[3] RFM12B USB light - Stick

<http://www.busware.de/tiki-index.php?page=RFM12BUSB>

[4] The Finch A Robot Designed for Computer Science Education

<http://www.finchrobot.com/>

[5] AD9833-based USB Function Generator

<http://tuomasnylund.fi/drupal6/content/ad9833-based-usb-function-generator>

[6] FlySight

<http://flysight.ca/>

[7] MIDI Products

<http://www.midi.org/aboutmidi/products.php>

[8] MIDI Note Player

<http://www.arduino.cc/en/Tutorial/Midi>

[9] MIDI Protocol Guide

<http://hinton-instruments.co.uk/reference/midi/protocol/index.htm>

[10] Updating the Atmega8U2 and 16U2 on an Uno or Mega2560 using DFU

<http://www.arduino.cc/en/Hacking/DFUProgramming8U2>

[11] Installing an Arduino Bootloader

<https://learn.sparkfun.com/tutorials/installing-an-arduino-bootloader/connecting-the-programmer>

[12] Programator Pocket AVR

http://www.robofun.ro/platforme/avr/programator_pocket_avr?utm_source=newsletter&utm_medium=email&utm_content=productLink&utm_campaign=CURS_EMAIL

[13] USBasp - USB programmer for Atmel AVR controllers

<http://www.fischl.de/usbasp/>

[14] Moco for LUFA

http://morecatlab.akiba.coocan.jp/morecat_lab/MocoLUFA.html

[15] AVRDUDE - AVR Downloader/UploaDEr

<http://www.nongnu.org/avrdude/>

[16] MIDI-OX

<http://www.midiox.com/>

[17] ATmega32U4

<http://www.atmel.com/devices/atmega32u4.aspx>

[18] Arduino Leonardo

http://www.arduino.cc/en/Main/arduinoBoardLeonardo?utm_source=newsletter&utm_medium=email&utm_content=productLink&utm_campaign=CURS_EMAIL

[19] Mouse and Keyboard libraries

<http://www.arduino.cc/en/Reference/MouseKeyboard>

[20] Teensy USB Development Board

<https://www.pjrc.com/store/teensy.html>

[21] Teensy2 USB Mass Storage with an SD card

<http://elasticsheep.com/2010/04/teensy2-usb-mass-storage-with-an-sd-card/>

[22] How Does Teensy 2.0 Compare With Arduino?

<http://www.pjrc.com/teensy/teensyduino.html>

[23] Making a USB flash drive HW Trojan

<http://jumpespjump.blogspot.ro/2013/01/making-usb-flash-drive-hw-trojan.html>