

Tommi Salomaa  
lopputyön dokumentaatio  
Musiikkiohjelmointi II, Tmu212CE  
Helsingin yliopisto, musiikkitiede  
22.5.2008

## **ESP-1: Midi-ohjattava äänilähde**

### **1. Johdanto**

ESP-1 (Experiment in Synthesizer Programming) on C-kielellä toteutettu midi-ohjattava monofoninen äänilähde, joka käyttää avoimen lähdekoodin PortMusic-kirjastoja (ks. PortMusic) äänen tuottamiseen ja midi-tapahtumien lukemiseen. Ohjelma tuottaa neljää perusaaltomuotoa: pulssi- saha- kolmio- ja siniaaltoja. Aaltomuodot lasketaan reaaliaikaisesti. Lisäksi pulssiaallon pulssinleveyttä voidaan muuttaa. Lähtökohtana on ollut PortAudion tutoriaalissa esitelty esimerkkiohjelma (ks. PortAudio tutorial).

Keskeisenä tavoitteena on ollut perehtyä matalan tason ääni- ja midiohjelmointiin PortAudio- ja PortMidi -kirjastojen avulla, ja erityisesti siihen, minkälaisia ohjelmointitekniikoita tarvitaan midi-ohjattuun äänen tuottamiseen ja tyypillisten midi-kontrolliominaisuuksien toteutukseen. Tästä syystä ohjelman ääntä tuottava osuus on pidetty yksinkertaisena, eikä myöskään käyttöliittymän toteutukseen ole kiinnitetty suurta huomiota. Äänen elävöittämiseksi ohjelmaan on kuitenkin toteutettu äänenvoimakkuutta säätelevä ADSR-verhokäyrä, sekä yksinkertainen vibrato. Verhokäyrän ja vibraton parametreja voidaan myös säätää ja ohjata midi-kontrollerien avulla.

## 2. Ohjelman rakenne ja toiminta

Kun ohjelma käynnistetään, se alustaa tarvittavat tietorakenteet, jonka jälkeen yritetään avata järjestelmän oletusarvoinen ääniulostulolaite `PortAudion OpenDefaultStream()`-funktiolla. Ohjelma hakee seuraavaksi midi-sisäänmenot jotka järjestelmästä on löydettävissä, ja pyytää valitsemaan näistä yhden. Mikäli ääniulostulovirta ja midi-sisääntulolaite on avattu onnistuneesti, ohjelma on toiminnassa. Ohjelman lopussa ääni- ja midivirrat suljetaan ja tietorakenteille varattu muisti vapautetaan.

Ohjelmassa on kolme säiettä: 1.) pääohjelma, 2.) `PortAudio`-säie joka vastaa äänen tuottamisesta ja äänidatan päivittämisestä midi-viestien perusteella, sekä 3.) säie, jossa luetaan mididataa. Audio- ja midisäikeet toimivat omissa callback-funktioissaan. Kommunikointi mididataa lukevan ja ääntä tuottavan säikeen välillä tapahtuu käyttäen `PortMidi`-kirjastoon kuuluvaa FIFO-jonorakennetta.

Perustietorakenteena on *SynthData*-rakenne, joka sisältää osoittimet *AudioData*- ja *MidiData* -rakenteisiin. *AudioData* sisältää äänen tuottamiseen tarvittavan tiedon, kuten ääniulostuloon lähetettävän varsinaisen äänidatan ja tiedon oskillaattorin taajuudesta ja vaiheesta. *MidiData* sisältää tietoa miditapahtumista. Tähän kuuluu tieto erilaisten kontrollerien tilasta sekä linkitetty lista, jonka avulla tallennetaan nuottitapahtumia. Listan avulla voidaan pitää kaikki alaspainetut nuotit muistissa. Jos esimerkiksi midi-koskettimistolla useampi kosketin on alaspainettuna, ja yksi vapautetaan, jää edellistä alaspainettua kosketinta vastaava nuotti soimaan.

Erilaisia midi-kontrollereita on myös toteutettu. Sitä, mihin parametriin tietty kontrolleri vaikuttaa, säädellään erityisen taulukon avulla. Taulukon osoittimena on jokin midi-kontrollerin numero, ja arvo kertoo mihin parametria kyseistä kontrolleria sovelletaan.

### 3. Testaus

ESP-1 on käännetty ja testattu onnistuneesti Linux (Debian/Ubuntu) -sekä Macintosh (OS X 10.5 Leopard) -ympäristöissä. Ohjelmaa on myös kokeiltu erilaisten midi-sovittimien kanssa. Keskeisenä havaintona oli, että suoraan USB-väylään kytkettävä koskettimisto lähetti dataa huomattavasti nopeammin kuin perinteiset midi-sovittimet. Midi-puskurin kokoa oli kasvatettava, jotta USB-koskettimisto saatiin toimimaan luotettavasti.

### 4. Kääntäminen ja käyttö

ESP-1 -ohjelman kääntämiseen tarvitaan mukana olevien lähdekooditiedostojen *espl.c*, *notelist.h* ja *notelist.c* lisäksi PortAudio- ja PortMidi-kirjastot, PortMidin mukana tuleva PortTime- kirjasto, sekä otsikkotiedostot *portmidi.h*, *portaudio.h*, *porttime.h* ja *pmutils.h*. Linuxissa linkitetään mukaan kirjastot *lrt*, *lpthread* ja *lasound*. OS X-ympäristössä tarvitaan seuraavat *kehykset* (engl. framework): *CoreAudio*, *AudioUnit*, *AudioToolbox*, *CoreServices*, ja *CoreMidi* (ks. PortAudio tutorial).

Kun ohjelma on avannut äänilaitteen, se listaa käytettävissä olevat midi-laitteet, joista valitaan yksi kirjoittamalla sen numero ja painamalla enter-näppäintä. Tämän jälkeen ohjelma on valmis vastaanottamaan midi-viestejä kaikilla kanavilla ja soittamaan ääntä niiden perusteella.

Ohjelmalle voidaan antaa komentoja komentoriviltä ohjelman ollessa käynnissä. Komennot ja parametrien arvot annetaan kokonaislukuina. Päävalikossa voidaan vaihtaa aaltomuotoa komennolla **1** ja säätää verhoikäyrää komennolla **2**. Ohjelma lopetetaan komennolla **0**.

Aaltomuodon vaihto tehdään antamalla halutun aaltomuodon numero 1-4 ja painamalla enter. Verhokäyrän säädössä ohjelmalle syötetään peräkkäin neljä lukua vastaamaan ADSR-verhokäyrän neljää arvoa *attack sustain, decay* ja *release*. Attack, sustain ja release -parametrien arvot ovat millisekunteina ilmaistavia aika-arvoja ja niiden arvoväli on 0–3000. Sustain-parametri kertoo prosentteina, millä tasolla maksimiarvosta äänenvoimakkuus on sustain-vaiheessa, ja se on asetettava välille 0–100.

Ohjelma soittaa ääntä midi-viestien perusteella midi-spesifikaation (ks. MIDI specification) mukaisten *note on* ja *note off* -viestien perusteella. Kosketusherkkyys vaikuttaa soivan nuotin voimakkuuteen. Myös äänenkorkeuteen vaikuttava pitch bend-säädin on toteutettu, samoin kanavakohtainen jälkikosketus, joka vaikuttaa vibraton syvyyteen. Lisäksi ohjelmaan on toteutettu erilaisia midi-kontrollereita. Taulukossa 1. luetellaan kontrollerit jotka vaikuttavat johonkin parametriin. Tällä hetkellä ainoa tapa muuttaa kontrollerien kohdeparametreja on asettaa ne `ctdest[]` -taulukossa `esp1.c` -tiedostossa ja kääntää ohjelma uudelleen.

Kontrollerin numero	Kontrollerin nimi	Parametri
1	modulaatiopyörä	pulssiaallon pulssinleveys
6	data-säädin	aaltomuodon valinta
7	äänenvoimakkuus	äänenvoimakkuus
22		verhokäyrä:,attack
23		verhokäyrä, decay
24		verhokäyrä, sustain
25		verhokäyrä, release
64	hold-pedaali	hold/sustain

**Taulukko 1.** Midi-kontrollerit

## 5. Lopuksi

ESP-1 ohjelman toteuttamisesta saatujen kokemusten perusteella PortMusic -kirjastot ovat suhteellisen helppoja käyttää, ja antavat tarvittavia välineitä kehittää ääni- ja midisovel-luksia jotka ovat siirrettäviä laiteympäristöstä toiseen. Tässä työssä ei tosin ole käytetty rajapintakohtaisia erityisominaisuuksia, joita PortAudiossa voidaan hyödyntää esimerkiksi paremman suorituskyvyn aikaansaamiseksi.

Ohjelma on perusrakenteeltaan toimiva, ja siihen voitaisiin helposti lisätä erilaisia ominai-suuksia, kuten useampia midi-kontrollereiden toteutuksia, sekä myös ääniominaisuuksia. Polyfoninen toteutus vaatisi kuitenkin laajempaa uudelleensuunnittelua.

Ohjelmassa on nykyiselläänkin paljon parannettavaa: esimerkiksi joidenkin midi-kontrollereiden, kuten pitch-bend -säätimen toteutukset ovat puutteellisia, samoin vibrato-toteutus. Taajuuden laskeminen kokonaisuudessaan vaatii parempaa suunnittelua. On otettava huomioon, että soivan äänen taajuuteen vaikuttavat sekä midi-nuotin numero, pitch bend-säätimen arvo sekä vibratoa tuottava koodi. Midi-tiedon lukeminen sekä sen soveltaminen audiodataan olisi myös ehkä syytä erottaa tarkemmin toisistaan: tällä hetkellä ne tehdään saman funktion sisällä.

### Lähteet:

PortMusic. <http://www.cs.cmu.edu/~music/portmusic/> (tarkistettu 20.5.2008).

PortAudio tutorial. <http://www.portaudio.com/trac/wiki/TutorialDir/TutorialStart>  
(tarkistettu 21.5.2008).

MIDI-specification. <http://www.borg.com/~jglatt/tech/midispec.htm> (tarkistettu 21.5.2008).

