LUND UNIVERSITY

# MIDI-controlled digital audio synthesizer

## Project proposal for EDAN85

By

Jaime Gancedo Rodrigo
Javier Gaztelumendi Arriaga
Berta Morral Escofet

Department of Electrical and Information Technology
Faculty of Engineering, LTH, Lund University
SE-221 00 Lund, Sweden

2017

# 1. Introduction

The objective of this document is to present our proposal for the EDAN85 (Design of Embedded Systems Advanced Course) project. We have decided to create a MIDI-controlled digital audio synthesizer, using the provided FPGA board, expansion peripherals for this board and some external hardware (a MIDI keyboard and, optionally, an external hardware controller).

# 2. Project overview and goals

Overall, the project consists of designing and successfully implementing an audio synthesizer on an FPGA making use of both one or several CPUs along with dedicated hardware designed in VHDL. The synthesizer's parameters will be MIDI (Musical Instrument Digital Interface) controlled thanks to an external keyboard compatible with this standard.

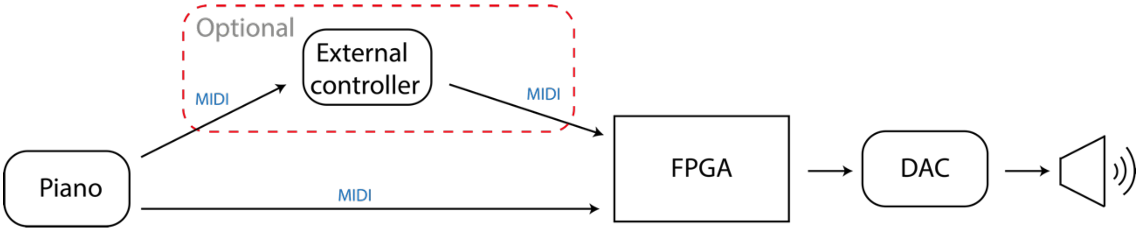


*Figure 1 - Top-level diagram of the system*

The basic features of our synthesizer, shown in Figure 1 will be:

- Three oscillators. Each with a different waveform. The suggested waveforms will be square, triangle and saw-tooth. Optionally, a fourth sine wave oscillator could be added. The system will be able to change the oscillators frequency as well as select which oscillator is being used at any instant.
- Poliphony. The system will be able to play several frequencies at the same time (several keys). This is important if the synthesizer is intended to be used as a real keyboard.
- Additive synthesis. The system will be able to mix two oscillators to create new waveforms. The main approach is that the user will be able to select which two oscillators and the amount of mix of each reach the audio output.
- MIDI-control. The system will be designed so that a user will only need to play the keyboard to generate MIDI messages which will then control the different parameters of our synthesizer.

If time permits, we would also like to incorporate the following features:

- Low frequency oscillator (LFO). An additional oscillator, operating at very low frequencies will be incorporated to modulate the former oscillators in one or more magnitudes: amplitude, phase or frequency.
- Filtering. Filters are used in subtractive synthesis and are very popular in synthesizers. It may be interesting to implement a low-pass filter with configurable cut-off frequency in the audio path.
- Additional effects. Different signal processing methods could be implemented in hardware as well to create reverb, delay or distortion effects to the audio signal. If more than one effect is implemented, the configuration of which effects are used at each

instant could also be added to the main control, so that the user can choose in real time what to do with the audio signal.

- External hardware controller. The MIDI keyboard we will use is limited on the signals that it can send. If we add several parameters to our system, we will need an additional external hardware controller with knobs and buttons/switches to configure them. If this extra-controller is added, the MIDI signal will go through it, in a daisy chain configuration.
- Auxiliar input (AUX). If one or more of the previous features are implemented, another possible feature would be to let the user connect an external audio source (a guitar, a smartphone, etc) and process its audio signal with the filter or the effects, routing it to the output, bypassing the synthesizer's oscillators.

# 3. Technical description of the project

## 3.1. Architecture

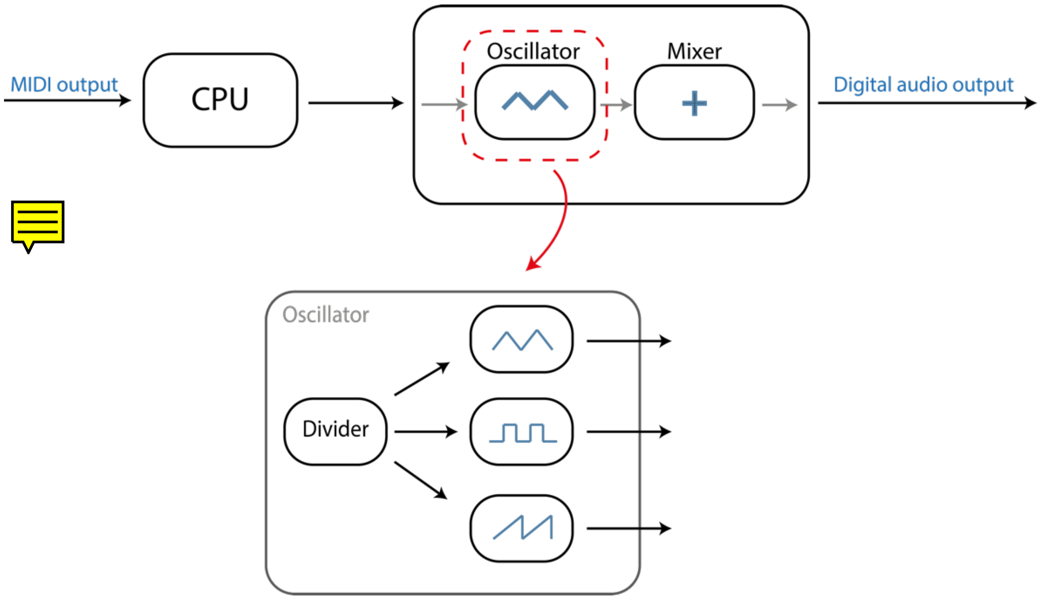


*Figure 2 - Suggested system architecture*

The architecture of the system, shown in Figure 2, is divided in two main parts: MIDI decoding and audio synthesis. The first is executed on the CPU, while the latter is performed by the custom hardware in the FPGA.

The main purpose of the MIDI decoding is to read the information sent by the MIDI controller and convert it to a set of target oscillator frequencies, waveforms and volumes. This MIDI interface should be fully compliant with the MIDI standard in order to ensure inter-operability with equipment from different manufacturers.

The audio synthesis part consists of several separately controlled oscillators, each of which can produce triangular, saw-tooth and square waves of an arbitrary frequency. The signal generated by the oscillators is then mixed to achieve polyphony and additive synthesis, this is, the creation of a new waveform by adding another two or more.

### 3.2. Specification

These are some key technical specifications that the synthesizer should fulfill:

- Output in the whole audible range (20Hz – 20kHz)
- MIDI standard compliant
- Triangle, saw-tooth and square signal output
- 10 note polyphony

# 4. Project plan

The project plan comprehends the six weeks that EDAN85 will have left since this proposal is presented.

- Week 1: Oscillators and audio output design.
- Week 2: Poliphony and additive synthesis design.
- Week 3: Additive synthesis testing and CPU based system design.
- Week 4: MIDI protocol and CPU software for decoding.
- Week 5: Testing of MIDI control for the control of system's parameters.
- Week 6: Wrap-up, bugfixes and final testing.

A Gantt diagram for this plan is shown in Figure 3.

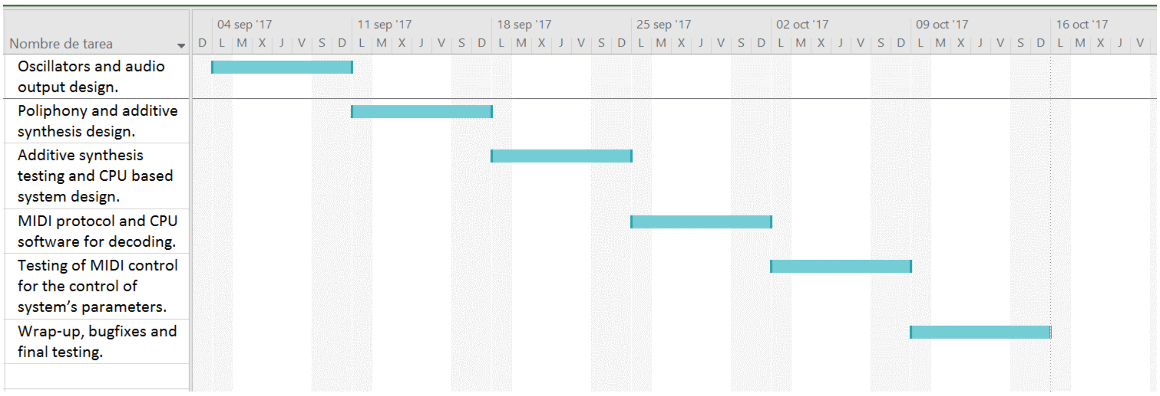


*Figure 3 - Gantt diagram for the project*