EECS 3201 – Final Project Report

FPGA Note Reader, UART Transmitter, and Real-Time Audio Playback

Ali Kamran

Member(s): Ali Kamran (solo)
Student Number: 219923184
YorkU Email: akamran@my.yorku.ca
Hardware: DE10-Lite FPGA + Arduino Uno (USB–Serial Bridge)

1. Project Overview

The FPGA generates selectable musical notes as a square wave from the 50 MHz clock and measures the note frequency by timing consecutive rising edges. The measured frequency is shown on the seven-segment display and sent over UART to a laptop program that plays the tone through the laptop speakers.

Top-Level Entity

The top-level Verilog module for the system is `Note_Reader`. This module interfaces directly with the DE10-Lite hardware (CLOCK_50, switches, push buttons, UART TX, LEDs, and the 6-digit seven-segment display). All internal components—including the UART transmitter, square-wave tone generator, frequency measurement unit, ASCII formatting logic, and debounced input handling—are instantiated inside this top-level module.

How combinational and sequential logic will be used

Combinational logic: pitch lookup table for one octave; octave scaling by shifts to compute the divider N; display encoding from binary to BCD for the seven-segment; ASCII formatting for UART output; simple multiplexers for display and status.

Sequential logic: divider counter that toggles the square-wave note every N cycles; free-running timestamp counter; synchronized edge detectors; period and high-time capture registers; integer division to compute hertz once per new period; seven-segment scan timer; debounced two-button interface; UART transmitter shift register with baud-tick generator.

Testable features

- Manual selection of 48 musical notes using on-board switches: 12 pitch classes (C through B) across four octaves (O3 to O6).
- Accurate frequency measurement using 50 MHz timestamping (rising-edge period capture). For example, A4 measures approximately 0440 on display.

- UART transmission at 115200 bps (8N1), sending ASCII lines such as `A4,0440,0441\r\n`.
- Real-time laptop audio playback using a custom C++ program with PortAudio.
- Freeze/mute controls, Force-A4 verification switch, LED status indicators.
- Arduino Uno acts as a USB–serial bridge between the FPGA and the laptop.

How to Run the Laptop Program

Once compiled (e.g., using `g++ note_player.cpp -lportaudio -o note_player`), run:

```
./note_player /dev/ttyACM0
```

Replace `/dev/ttyACM0` with the port used by your Arduino Uno.

References

- Intel DE10-Lite User Manual (Cyclone V FPGA documentation)
- UART 115200 8N1 standard conventions
- PortAudio API reference
- Arduino Uno ATmega328P serial bridge documentation
- Course lecture notes and lab materials (EECS 3201)