16-Steps Sequencer and Synthesizer

Zhuowen Lin & Jinyue Guo

11510818 & 11510478

Information Engineering, SUSTech

Speech Outline

- Inputs & Outputs
- Components of the circuit
- Result demonstration
- Further enhancements

Inputs & Outputs

Inputs

- Switches
  - Each of the 16 switches is assigned to a note at a specific time.
  - When the time passes it, a note of the selected frequency will play.
- Buttons
  - BTNR: Save: assign any switch that is in the on position with the current frequency.
  - BTNU: Switch frequency: rotate through the 16 frequencies.
  - BTNC: Play/Pause
  - BTNL: Reset: erase all assigned tones.

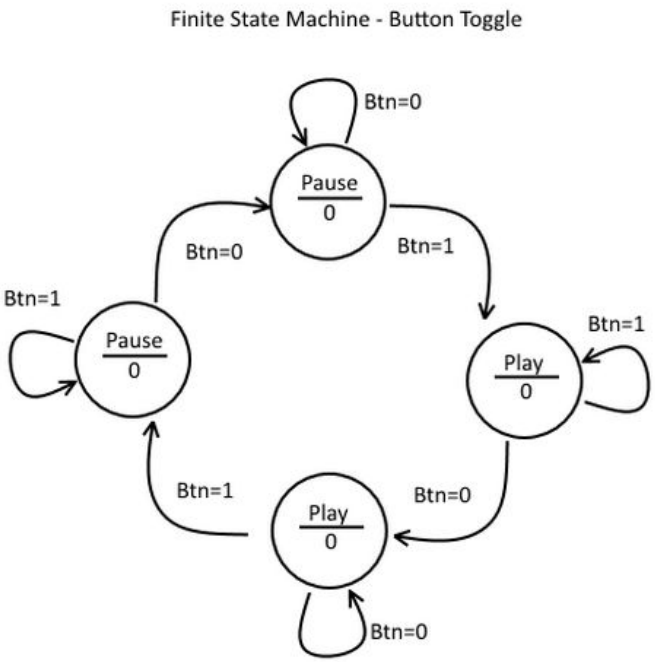
Outputs

- LEDs
  - LEDs above each switch.
  - Light up when a tone is playing on that particular switch.
- Seven Segment Display
  - Show which frequency is being worked with.
  - Rotate from 0 to F.
- Speaker
  - Sound wave at different frequencies.

Components of the Circuit

Button Toggle – Play & Pause

- FSM
  - Toggle between two states exactly one time when the button is pressed



Button Debouncer

- Button bounce
  - Switch between states in FSM
- Debouncer
  - By Scott Larson of Digi-key
  - 10.5 ms stable input time

Clock Dividor – BPM setting

- Convert 100 MHz to lower frequencies
  - Used to set BPM

Counter – Tone / Switch Tracker

- Keep track of the tone
  - Speaker knows which tone to be played
  - Seven segment display knows which number to be displayed
- Keep track of the switch
  - Speaker knows which switch position to be played
  - Counter increasing speed: BPM
- T flip-flop
  - 4 bits ←→ 4 T flip-flops
  - Least significant bit: toggle each time
  - The other bits: toggle when all of bits below them are equal to one.

D Flip-Flop

- Remember the tone of 16 switches

Seven Segment Display

- Course Lab 3
  - Hexadecimal number

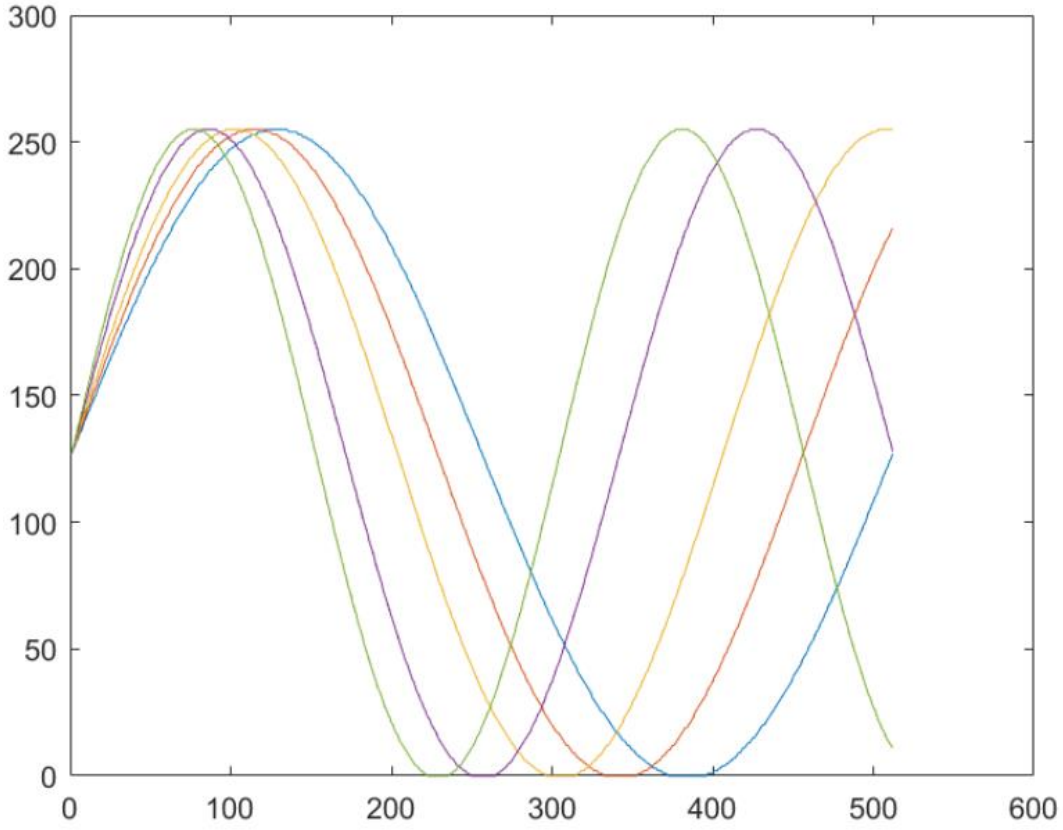
Square Wave Generator

- Change frequency dividing factor

Number	Tone	Max Count	Number	Tone	Max Count
0	0	0	8	d2	85131
1	a	227272	9	e2	75843
2	c1	191114	10	g2	63776
3	d1	170262	11	a2	56818
4	e1	150372	12	c3	47778
5	g1	127552	13	d3	42566
6	a1	113636	14	e3	37922
7	c2	95556	15	g3	31888

Sine Wave Generator

- Waveform dictionary
  - 512-length arrays
  - Respective frequencies
- Value loader
  - Index: read value
  - Different max values



Sine Wave Generator - frequency

- Octave freq. divider
  - Save memory space
  - Use only Five distinct dictionaries
- Physical freq. divider
  - Divides oscillator clk into real sine wave clk

A_4	440.00
$A^{\#}_4/B^{b}_4$	466.16
B_4	493.88
C_5	523.25
$C^{\#}_5/D^{b}_5$	554.37
D_5	587.33
$D^{\#}_5/E^{b}_5$	622.25
E_5	659.25
F_5	698.46
$F^{\#}_5/G^{b}_5$	739.99
G_5	783.99
$G^{\#}_5/A^{b}_5$	830.61
A_5	880.00

Sine Wave Generator - Structure

- Waveform dictionary
- Physical freq. divider
- Pre-loader
  - Input tone (4bit)
  - Load dictionary
  - Maximum index
  - Octave div coefficient
- Octave freq. divider
- Output value loader

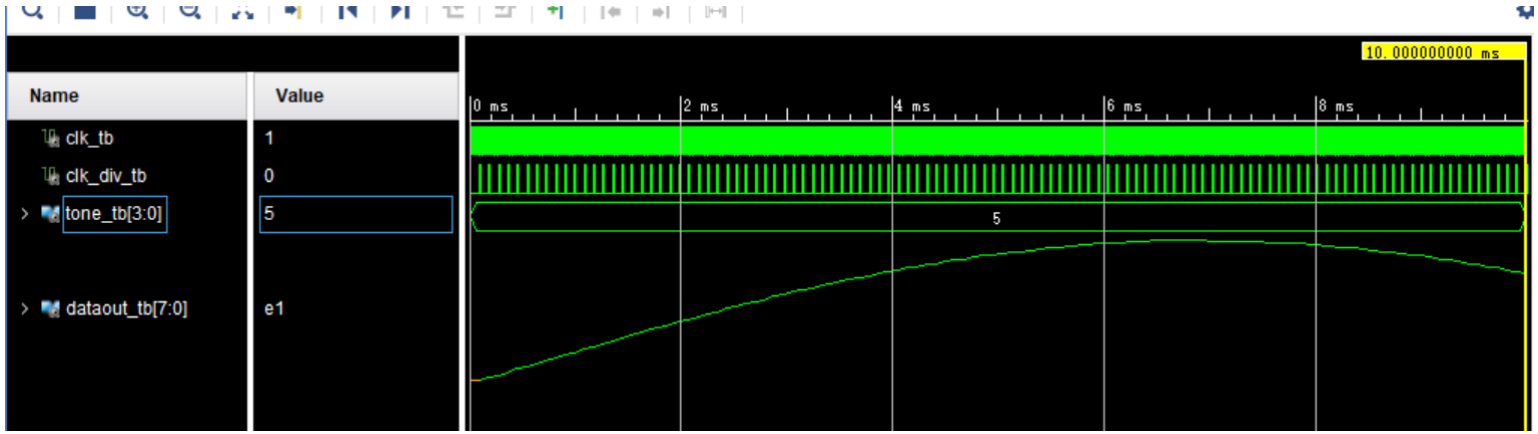
- Input:
  - Clk: std_logic
  - Tone: std_logic_vector (3 downto 0)
- Output:
  - Wave_out: std_logic_vector(7 downto 0)

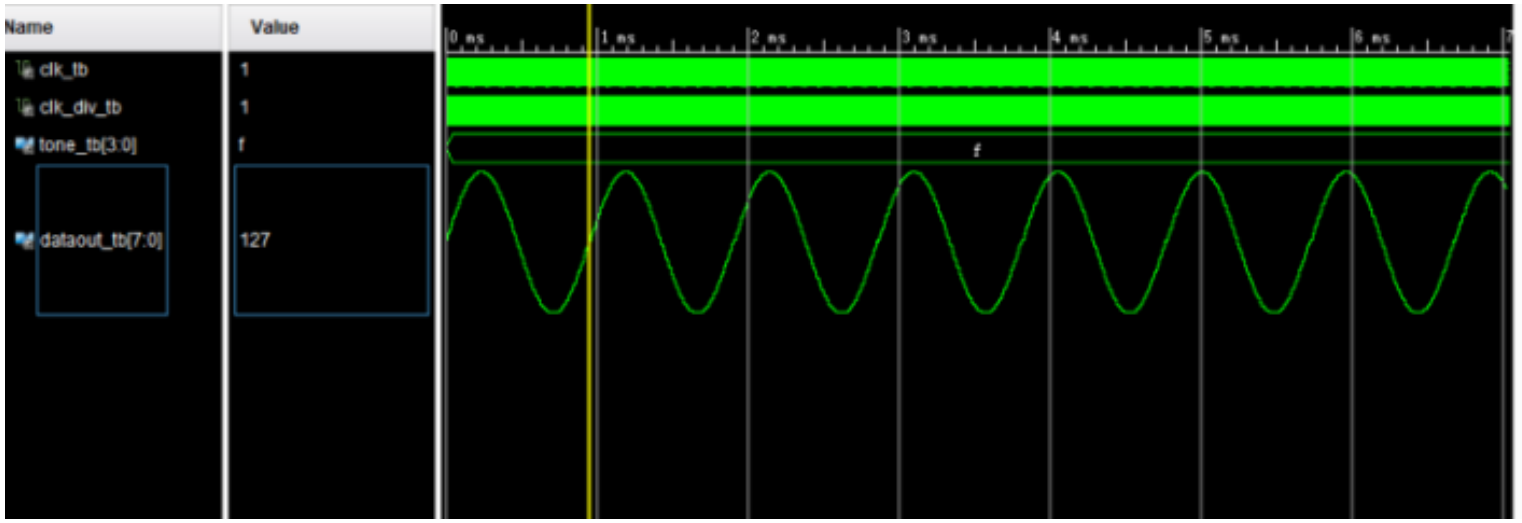
Sine Wave Generator - Structure

- Why having two freq. dividers?

- Separate the waveform generating process
- Easier to create new dictionaries

Sine Wave Generator





PWM Encoder

- Nexys 4 DDR has a built-in PWM (Pulse Width Modulation) decoder, which is a low-pass filter.

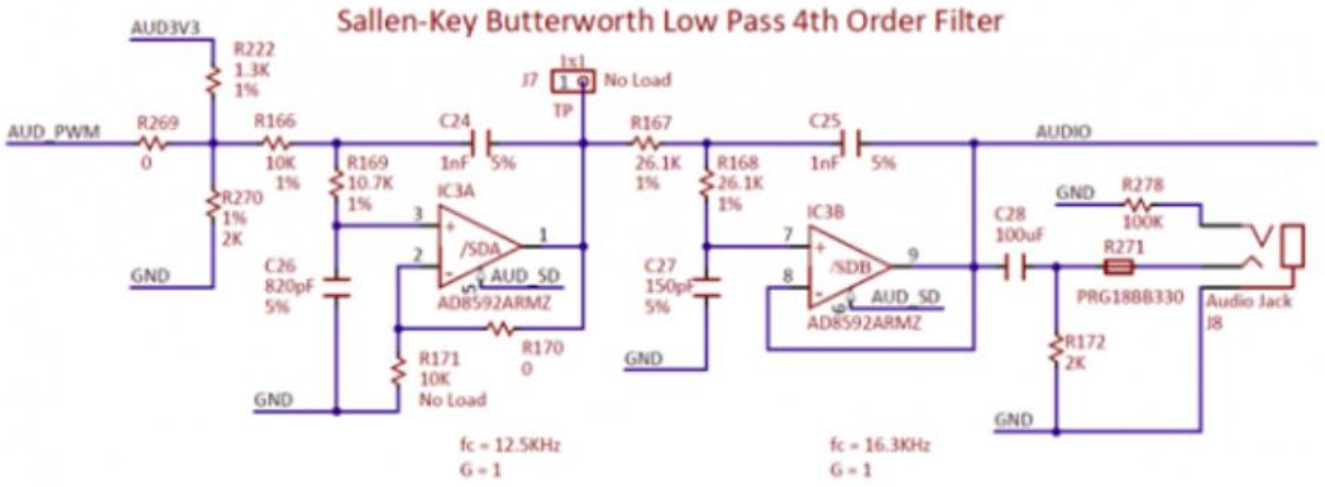


Figure 29. Sallen-Key Butterworth Low-Pass 4th Order Filter.

PWM Encoder

- Input: wave_in : std_logic_vector (7 downto 0)
- Calculate Duty Cycle threshold, according to PWM frequency
- Counter: smaller than threshold, output ‘1’; else ‘0’
- Output: std_logic

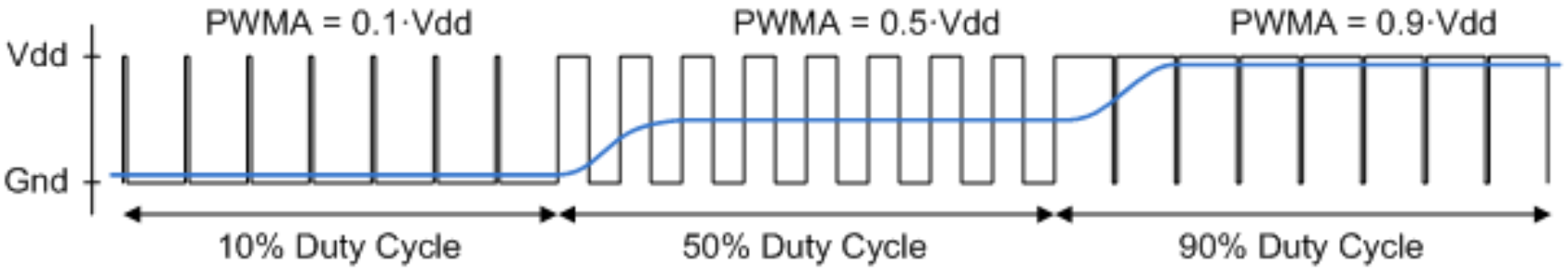
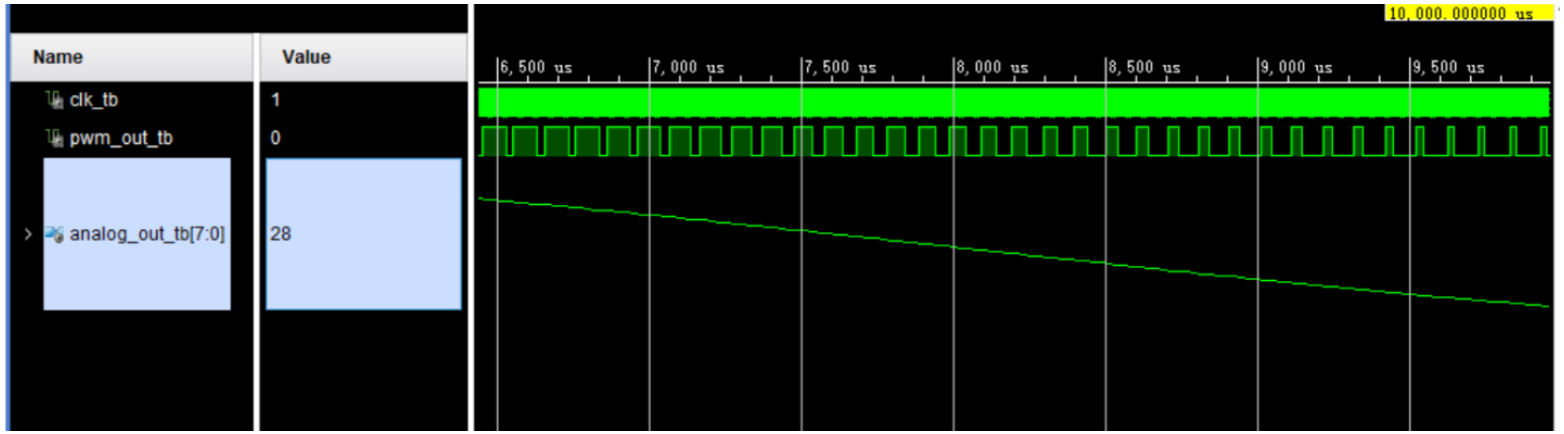
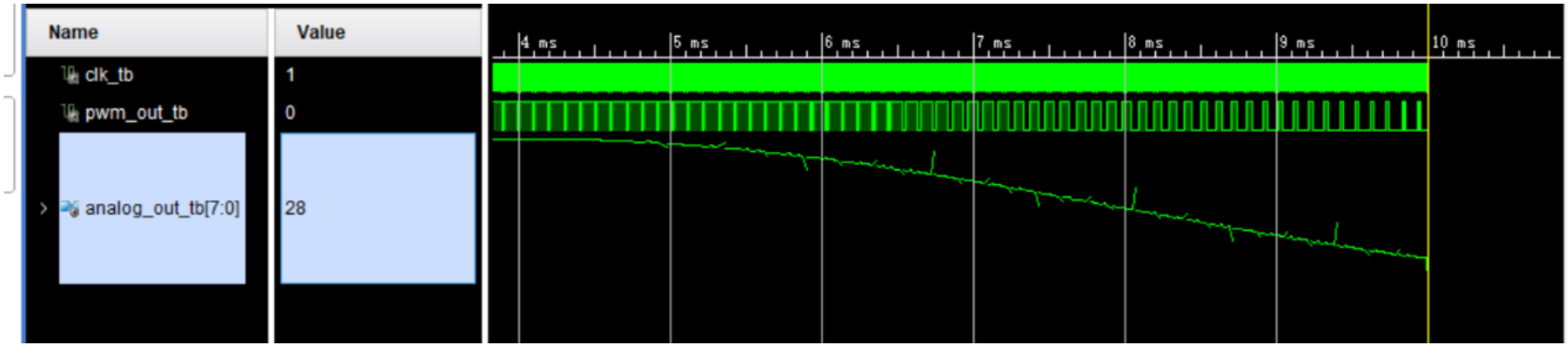


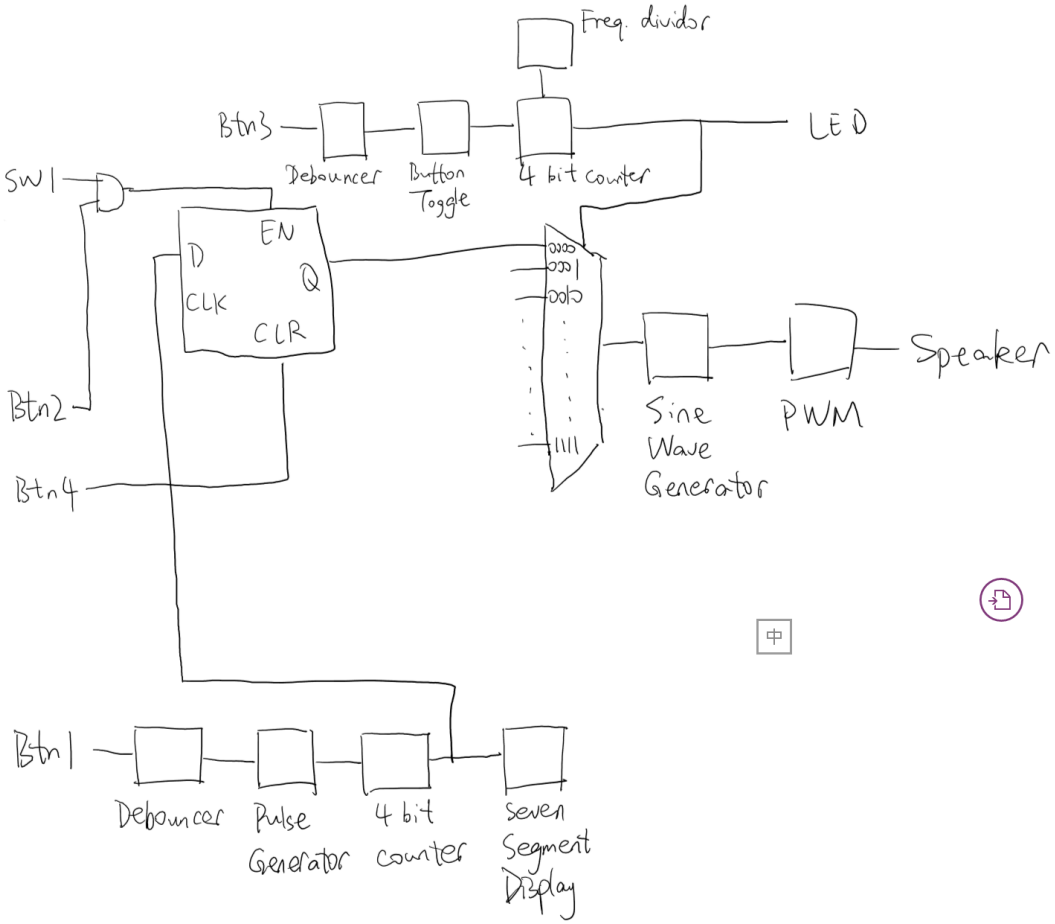
Figure 32. Representation of a PWM integrator producing an output voltage by integrating the pulse train.

PWM Encoder


Name
Value
clk_tb
1
pwm_out_tb
0
analog_out_tb[7:0]
28
10,000.000000 us
6,500 us
7,000 us
7,500 us
8,000 us
8,500 us
9,000 us
9,500 us


Name
Value
clk_tb
1
pwm_out_tb
0
analog_out_tb[7:0]
28
4 ms
5 ms
6 ms
7 ms
8 ms
9 ms
10 ms

Connecting Components



Result Demonstration

Further Development

- Better volume/ DA output solution
- Beats synthesizer
- Synchronization ports
- Selectable timbre (square, sine, toothsaw, etc.)
- Changeable BPM

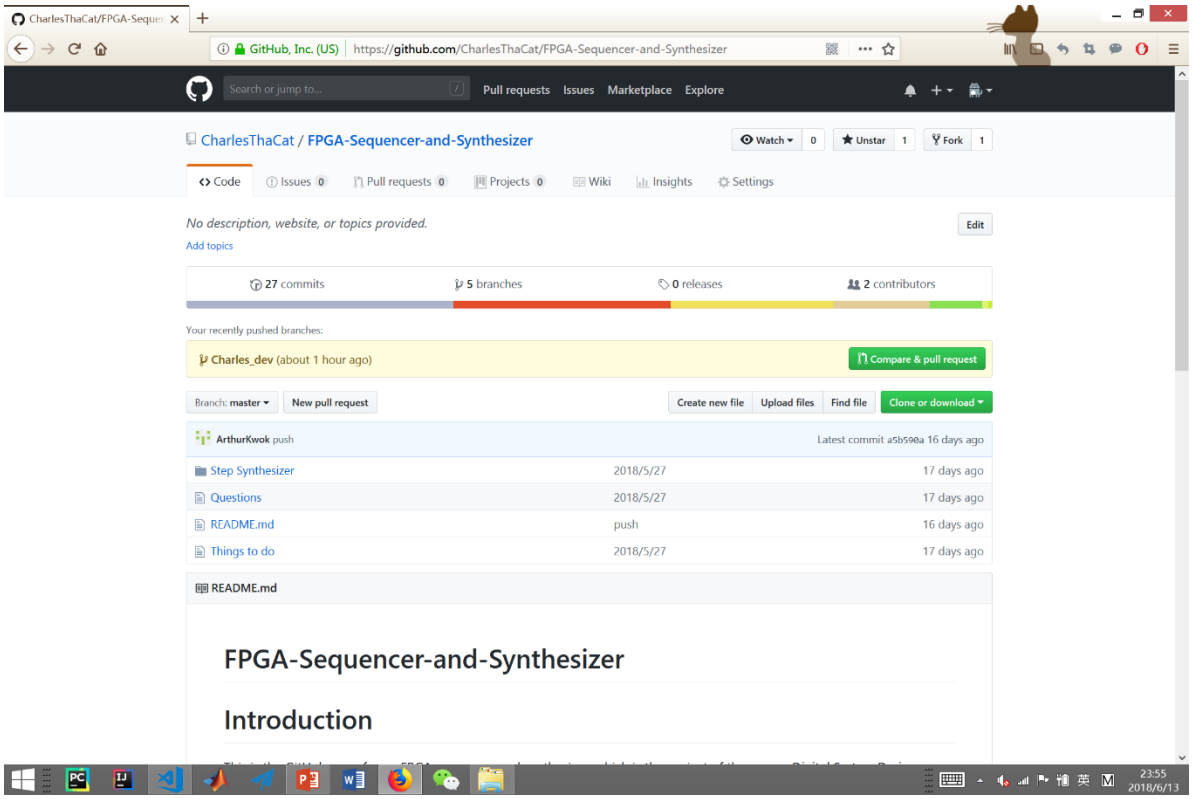
Reference

- Frequencies for equal-tempered scale, A4 = 440 Hz
  http://pages.mtu.edu/~suits/notefreqs.html
- Nexys 4 DDR Reference Manual
  https://reference.digilentinc.com/reference/programmable-logic/nexys-4-ddr/reference-manual
- PWM Audio Tutorial using a FPGA
  https://www.youtube.com/watch?v=4byHVqXD-UI
- Class Slides

Q&A

GitHub Page

- https://github.com/CharlesThaCat/FPGA-Sequencer-and-Synthesizer



Thank You!