Shane Skikne
Midterm Computer Architecture
Due 11/15/15

In my copied work, the printer keeps adding lines that make it very confusing and I can't figure out why so I have included pictures as well. Sorry about

Summary: This product is a bike light that cycles between 4 different modes, Off, On, Blink and Dim. It has a single button and a single light. Mode is switched when the button is released button.

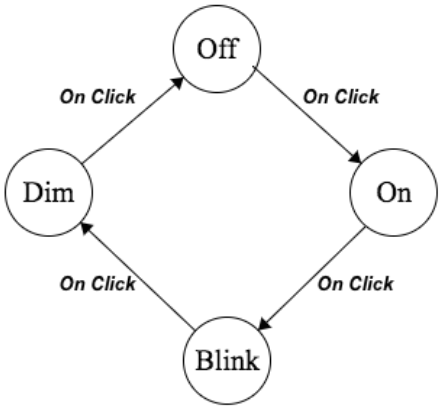
Input: There is a single button the user can press. The user can press and hold the button with no affect, but the mode will be changed as soon as the user releases the button. If the user presses the button repeatedly, the device will cycle through its modes.

Output: There is a single light. For each mode, the light will be controlled using a different pattern. The light needs to be able to turn on and off very quickly. While the light can switch modes as soon as the button is released, it takes a short period of time for the light to start working correctly.

Mode Description:

Off: The light stays off the entire time.
On: The light stays on the entire time.
Blink: The light turns on and off for an equal amount of time. Each cycle of On/Off should last approximately .25 seconds (so 4Hz).
Dim: The light appears to be on the entire time, but at a dimmer brightness (50% of the normal brightness). This is done using pulse width modulation, where the light is cycled On/Off at 32,768 Hz, so it simply appears to be dimmer.

State Machine:



Wave Form:

The waveform includes a simplification in order to make it possible to see. It is very difficult to have the wave form with the blink and the dim on the same waveform, because of their vastly different frequencies. Therefore, the frequency of the blink mode is kept at 4Hz, while the frequency of the Dim is made to look like it is 12Hz, when it is really significantly higher. In the real device, the dim feature should have a much shorter period.

Components

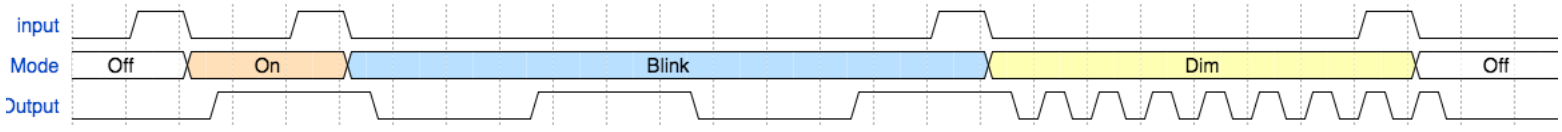
N-Adder Input Conditioner
Specification: The input conditioner takes an input and conditions it to control for any rapid changing in the input. The amount of time until the input must be different from the conditioned input is based on the number of Adders, N, and the clock cycle with the relationship of

$$\frac{1}{Freq\ of\ Clock\ Cycle} \times 2^N$$

Inputs: Clock and Input to be conditioned
Outputs: Conditioned Input, Positive edge of the conditioned input, Negative Edge of the Conditioned Input
Schematic: See Attached.
Cost: 38+N(27).

Adder
Specification: This adds 2 bits and returns the sum and the carry out.
Inputs: Bit 1 and Bit 2, the 2 bits that can be added.
Outputs: Sum and Carry Out
Schematic: See Attached.
Cost: 11.

Blinker:
Specification: When this component is enabled, it outputs a square wave with frequency 4Hz.
Inputs: Enable, which enables to the output.
Outputs: Out which is 0 when Enable is 0 and a 4Hz Square Wave when Enable is 1.
Schematic: See Attached.



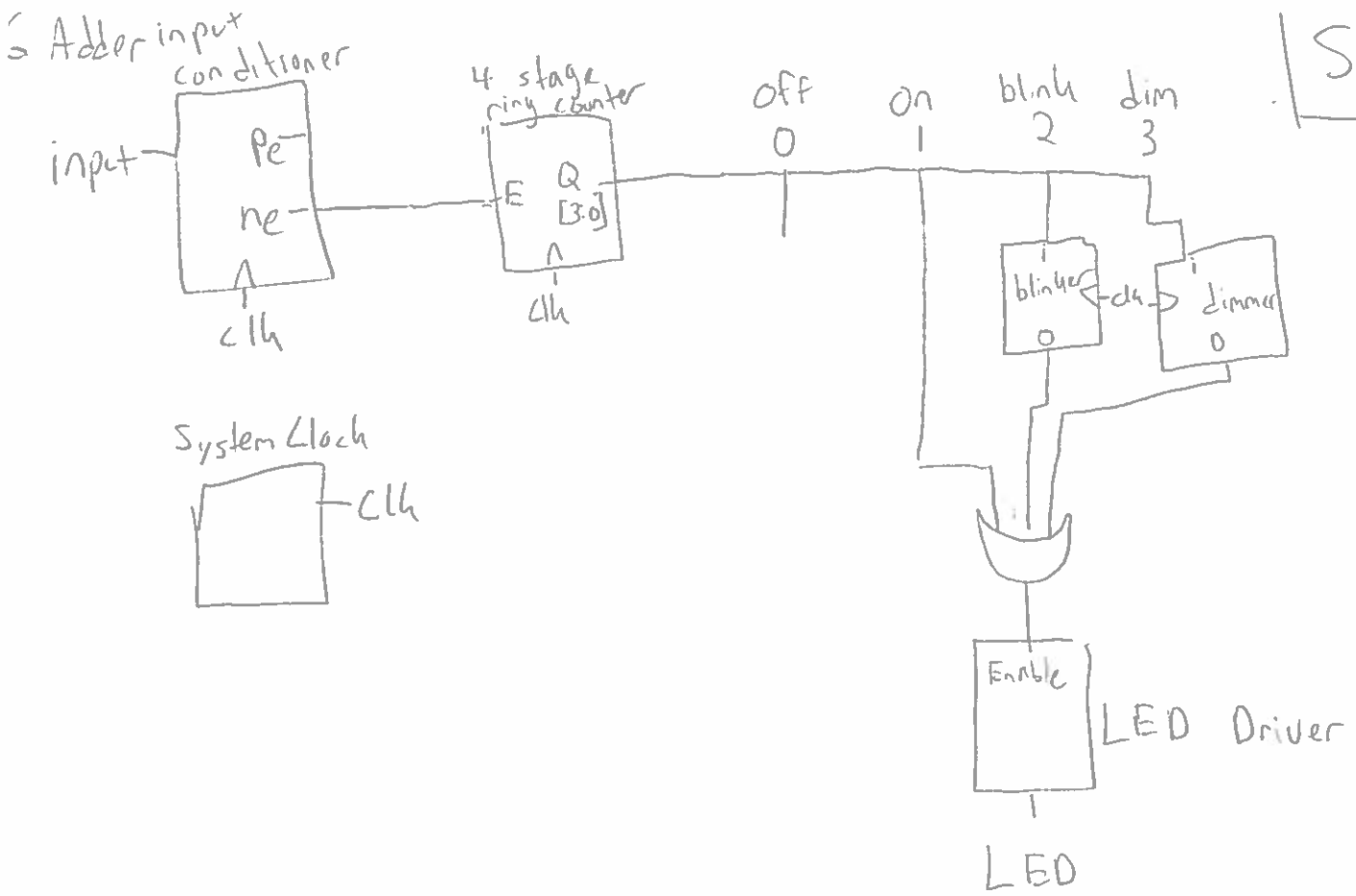
Cost: 12*27+38+4 = 366.

Dimmer*:
Specification: When this component is enabled, it outputs a square wave with frequency 32,768Hz.
Inputs: Enable, which enables to the output.
Outputs: Out which is 0 when Enable is 0 and a 32,768Hz Square Wave when Enable is 1.
Schematic: See Attached.
Cost: 17.

*I also could have just used the clock as my square wave, which would save me 14 cost.

Total Cost: 2+204+83+366+17+4+211 = 887. See Schematic for details.

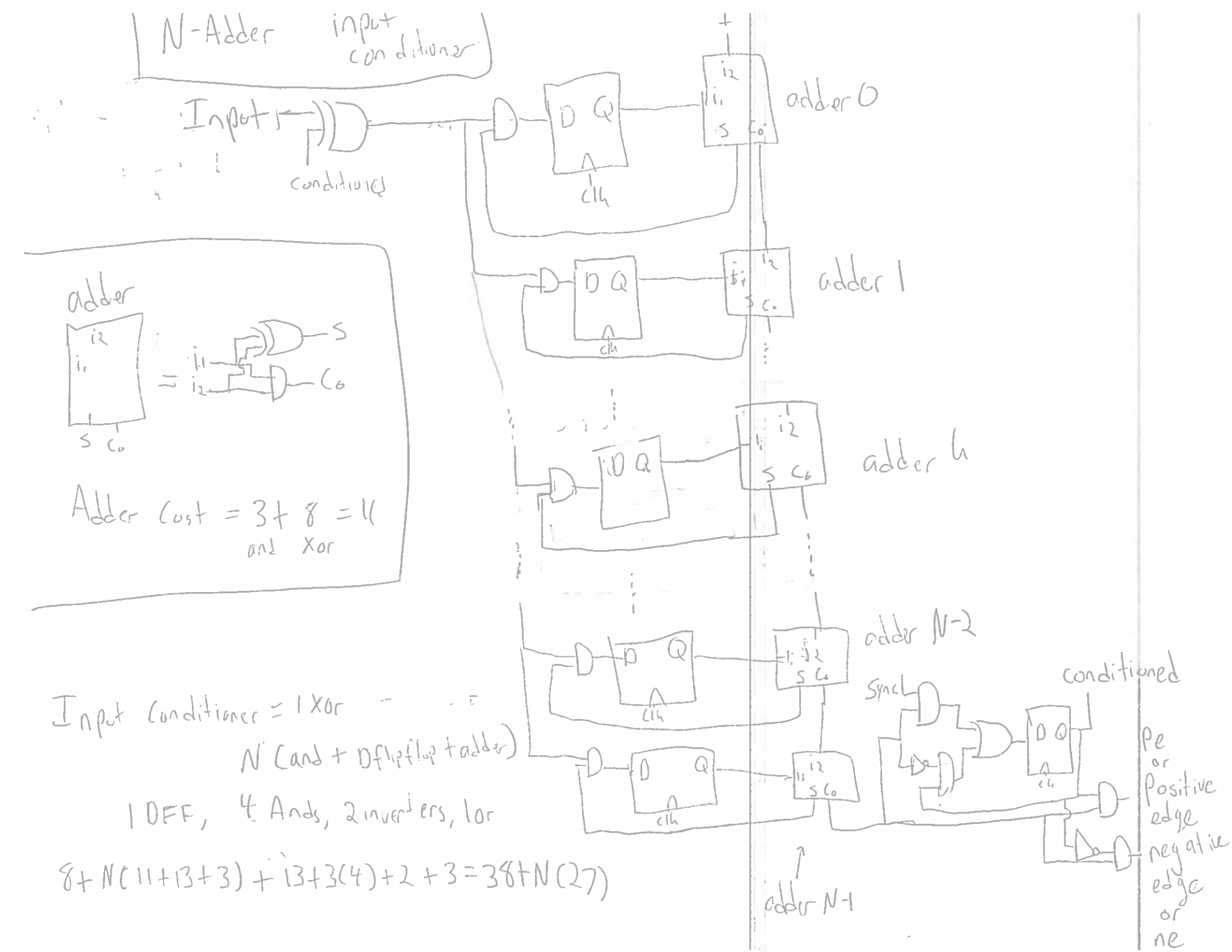
Schematic



Total Cost = 2 + 204 + (84 - 1) + 366 + 17 + 4 + 211 = 887

System Clk — 6 Adder input conditioner — 4 stage ring counter — blinker — dimmer — 3 input OR — Led Driver

N-Adder input conditioner



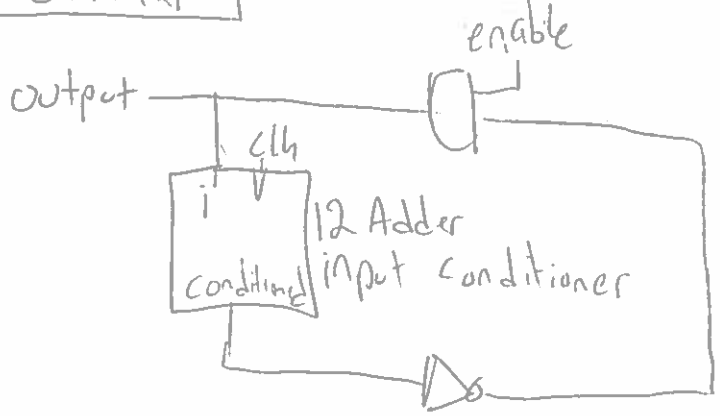
adder

Adder Cost = 3 + 8 = 11
and Xor

Input Conditioner = 1 Xor
N (and + Dflipflop + adder)
1 DFF, 4 Ands, 2 inverters, 1 or

8 + N(11+13+3) + 13+3(4)+2+3 = 38+N(27)

Blinker



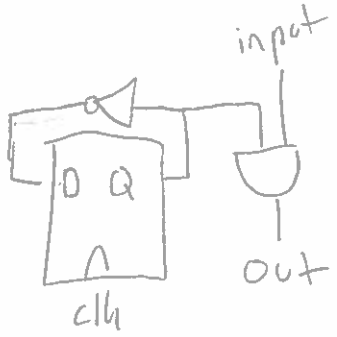
Cost

12 x 17 + 38 + 1 + 3 = 366

12 Adder input conditioner

inverter and

Dimmer



1Dff + 1 inverter + 1 And

13 + 1 + 3 = 17

Block Diagram