**State Machine Design**
An old TV remote control and TV only supports 2-digit channels: 00-99. Normally, to change the channel we would have to hit two buttons: 38, first 3 then 8, and as soon as you hit the second button it should change the channel. However, for channels 0-9 the remote should allow you to just enter 1-digit and if another button is not pushed soon afterwards should cause the channel to be changed. Implement a state machine that can indicate when the channel should be changed. After it is changed, restart waiting for a push.

Have a single input PUSH and single output CHANGE. If you get the 1st push but then don't get the second push within 2 cycles, just change the channel no matter what. Use four states: 0PUSH, 1PUSH, WAIT, CHGCHAN.

Adder and Combinational Design

Suppose team X and team Y play a game where their scores range 0-7 decimal. Team X is much better than team Y so they give Y a handicap: to truly win, team X must score 5 points more than team Y. Design a circuit that will produce a signal: XLoses given the two 3-bit input numbers X[2:0] and Y[2:0] representing the scores of each team.

X2 X1 X0   Y2 Y1 Y0

B3 B2 B1 B0   A3 A2 A1 A0

C4   *4-bit Binary Adder*   C0

S3   S2   S1   S0

XLoses
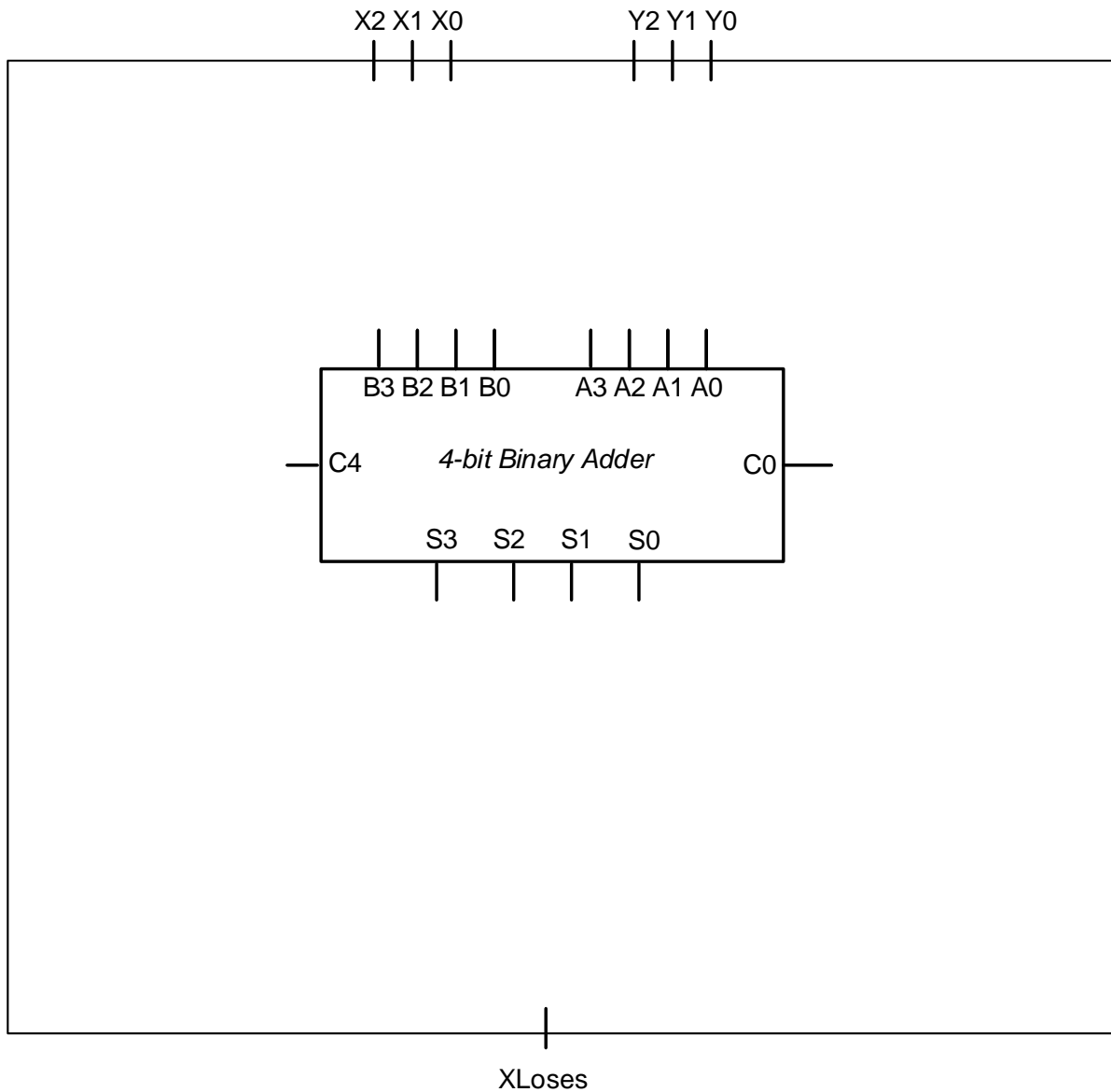
Determine a prescalar, OCR0A value, and ISR for an 8-bit timer generate at 400Hz square wave for 3 seconds on PD1. Recall the Arduino runs at 16MHz and valid prescalars are: 1, 8, 64, 256, 1024 (choose the smallest prescalar that works).

```
ISR(TIMER0_COMPA_vect)
{




}
```

Show how to implement this flip-flop with load enable by determining the values to program into an FPGA's configurable logic block shown to the right.

3x1 CLB

8x1 Mem.

Q — A_0 0
D — A_1 1
EN — A_2 2
3
4
5
6
7
D_0

D
CLK
Q

1 0

Q

0
Y
D — 1 S
EN
CLK
D Q Q