I. Learner Objectives:

At the conclusion of this lab, participants should be able to:

- Read and analyze a set of requirements for a real-world application
- Apply read and write operations to ports
- Manipulate peripheral modules connected to the Digilent Cerebot MX4cK Board
- Compose a small embedded C language program
- Comment a small embedded C language program

II. Prerequisites:

Before reading this lab, participants should be able to:

- List and describe registers belonging to the MIPS32 architecture
- Summarize the material found in the PIC32MX MCU unit
- Diagram and annotate the architecture associated with the PIC32MX460F512L microcontroller

III. Keywords:

Port, I/O register, peripheral module, SYSCLK, PBCLK

IV. Pre-lab (in-lab):

Non for this week!

V. Lab:

What to complete:

In this lab you will use MPLAB and one provided PIC32 C library to create a basic binary up-down counter that displays the current value of the counter on the LEDs. You will use the on-board buttons of the Cerebot MX4cK to start, stop, and clear the counter. The current 3-bit result of the counter should be displayed on the on-board LEDs. The state transition diagram below provides a partial view of the required events and states. You must fill in the rest of the transitions and states. Once you have completed it, use it along with the below requirements to design and implement an appropriate up-down counter. You may include the following in your main.c file to configure SYSCLK and PBCLK.

```
// SYSCLK = 80 MHz (8 MHz Crystal/ FPLLIDIV * FPLLMUL / FPLLODIV)
// PBCLK = 40 MHz
// Primary Osc w/PLL (XT+,HS+,EC+PLL)
// WDT OFF
// Other options are don't care

#pragma config FNOSC  = PRIPLL // Required
#pragma config FPLLMUL  = MUL_20
```

```
#pragma config FPLLIDIV  = DIV_2
#pragma config FPBDIV  = DIV_2 // Divide SYSCLK by 2 ==> PBCLK
#pragma config FPLLODIV  = DIV_1
```

Recall the SYSCLK is the frequency at which the CPU will operate. The I/O ports operate based on SYSCLK. Peripherals such as timers and UARTs operate based on the PBCLK. For this lab, you most likely will not be concerned about the PBCLK. Please just learn how to set the SYSCLK and the PBCLK.

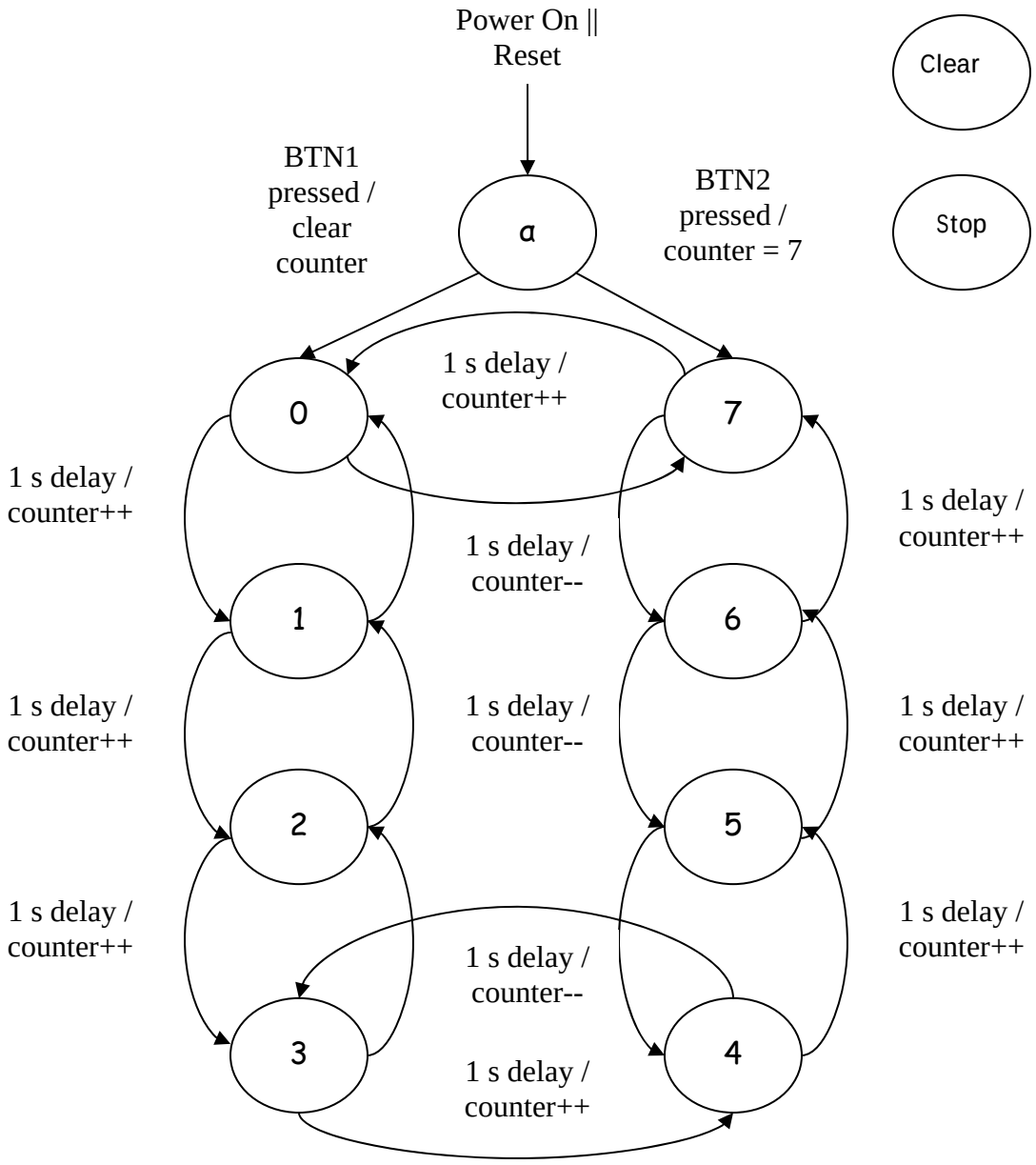
Requirements:

- Pressing on-board BTN1 must start the up counter (counts 0 → 7)
- Pressing on-board BTN2 must start the down counter (counts 7 → 0)
- Pressing BTN1 and BTN2 together once must stop the counter
- Pressing BTN1 and BTN2 together twice must clear the counter (counter = 0)
- Complete the partial state transition diagram
- Answer the questions provided below in section VI.

Hardware Required:

- 1 - Digilent Cerebot MX4cK Embedded Controller Board
- 1 – USB A -> micro B programming cable

Partial State Transition Diagram for Up-down Counter:



VI. Questions:

Please include the answers to the following questions in your formal lab write-up!

1. Analyze the disassembly listing for your program. Indentify and describe the function of three different registers.

2. Analyze your complete state transition diagram. Can you reduce the number of states visualized? If yes, show an updated state transition diagram illustrating your changes.

3. Any comments you would like to make to spice-up or improve this lab for the future!