# Contents

**Contents** **1**

When clicking on a link in Adobe use alt+arrow left to return to where you started.

# Laboratory 1

# Stopwatch Control Unit

## 1.1 Objective

The objective of this lab is to unify the stopwatch datapath and control units into a working design that runs on the Cyclone V GX development board.

## 1.2 Discussion

From the previous lab, you should be familiar with the operation of our stopwatch. Briefly, our stopwatch allows a user to measure elapsed time and lap times of a competitive events. Our stopwatch measures time in increments of a tenth of a second, unit second and tens of seconds. Control input comes from 2 buttons called S1 and S2 according to the finite state machine shown in Figure 1.1.

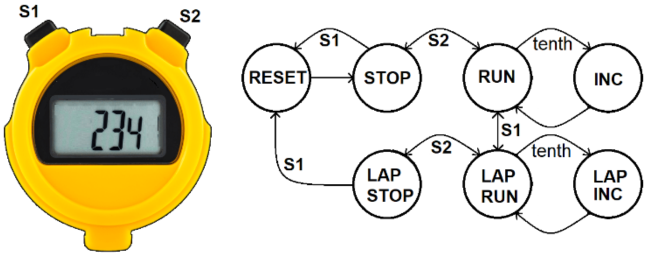

Figure 1.1: A digital stopwatch gets its input from 2 buttons and displays its output on a 7-segment display. The behavior of the stopwatch can be described by this finite state machine (FSM).

Figure 1.2 shows how you will implement the idea presented in Figure 1.1 using the FGPA development board.

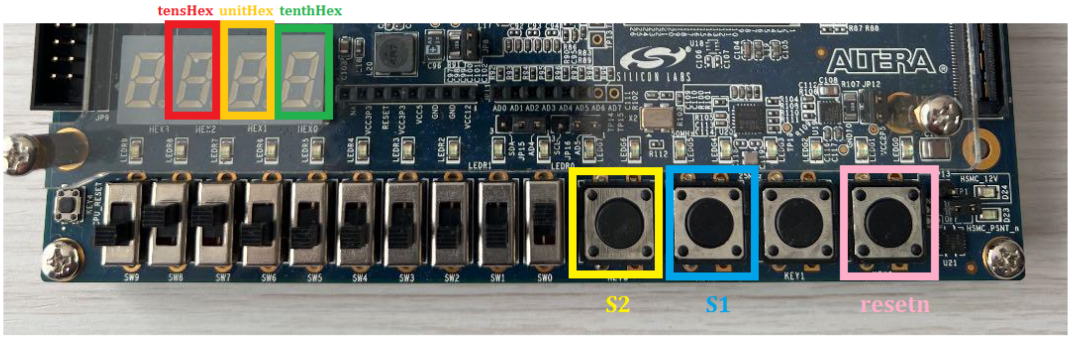

Figure 1.2: The stopwatch user interface as implemented on the FPGA development board.

Two of the Cyclone V buttons will act as the stopwatch buttons and 3 of the 7-segment displays will show the time. The reset signal is connected to a button and the 50MHz clock is on the Cyclone V GX development board but does not require any user interaction.

## 1.3 Stopwatch architecture

In the previous 2 labs you have created the datapath and control unit for the stopwatch. Using these 2 components as building blocks, the architecture for the stopwatch, shown in Figure 1.3, is almost trivial; the Verilog file for the stopwatch contains 2 component instantiations of the datapath and controlUnit.

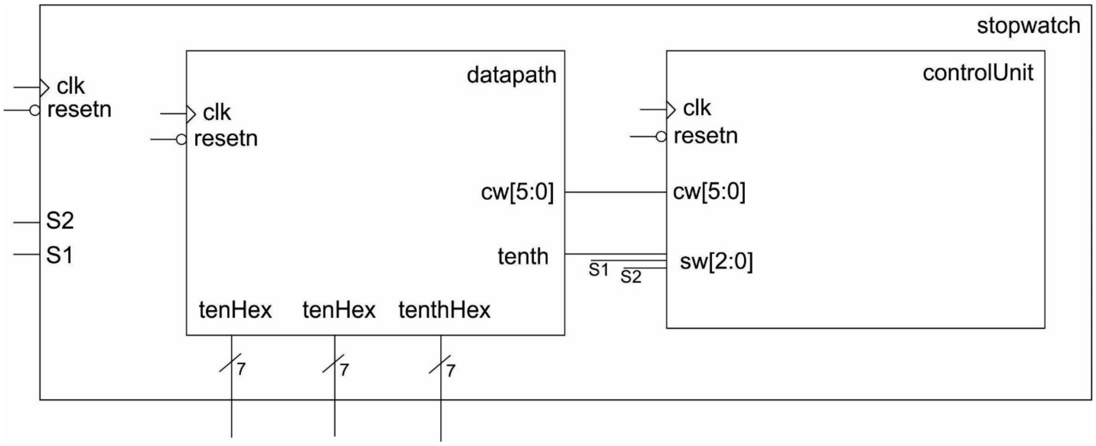

Figure 1.3: The architecture for the stopwatch consists of a datapath and controlUnit.

The only minor complication is combining the tenth output from the datapath with the S2 and S1 signals coming in from the Cyclone V GX buttons into a 3-bit signal sent to the status word input of the control unit.

Use the starter code provided on Canvas to complete the stopwatch module. While you are at it, download the stopwatch testbench provided on Canvas, and make the testbench the top-level entity.

## 1.4 Stopwatch Simulation

Before you download your completed control unit to the Cyclone V boards, you are going to perform extensive simulations to uncover as many bugs as possible. Errors are much, much easier to find in a simulation. First you will need to understand what the testbench simulation is supposed to do.

The timing diagrams in Figure 1.4 show the S1 and S2 signals as they are manipulated by the testbench. The tenth signals will be asserted by your datapath but are included to help you. Your task is to fill in the symbolic name of the state that the FSM is in as well as the clkCount value, the mod10Counter outputs and the values displayed on the 7-segment displays.

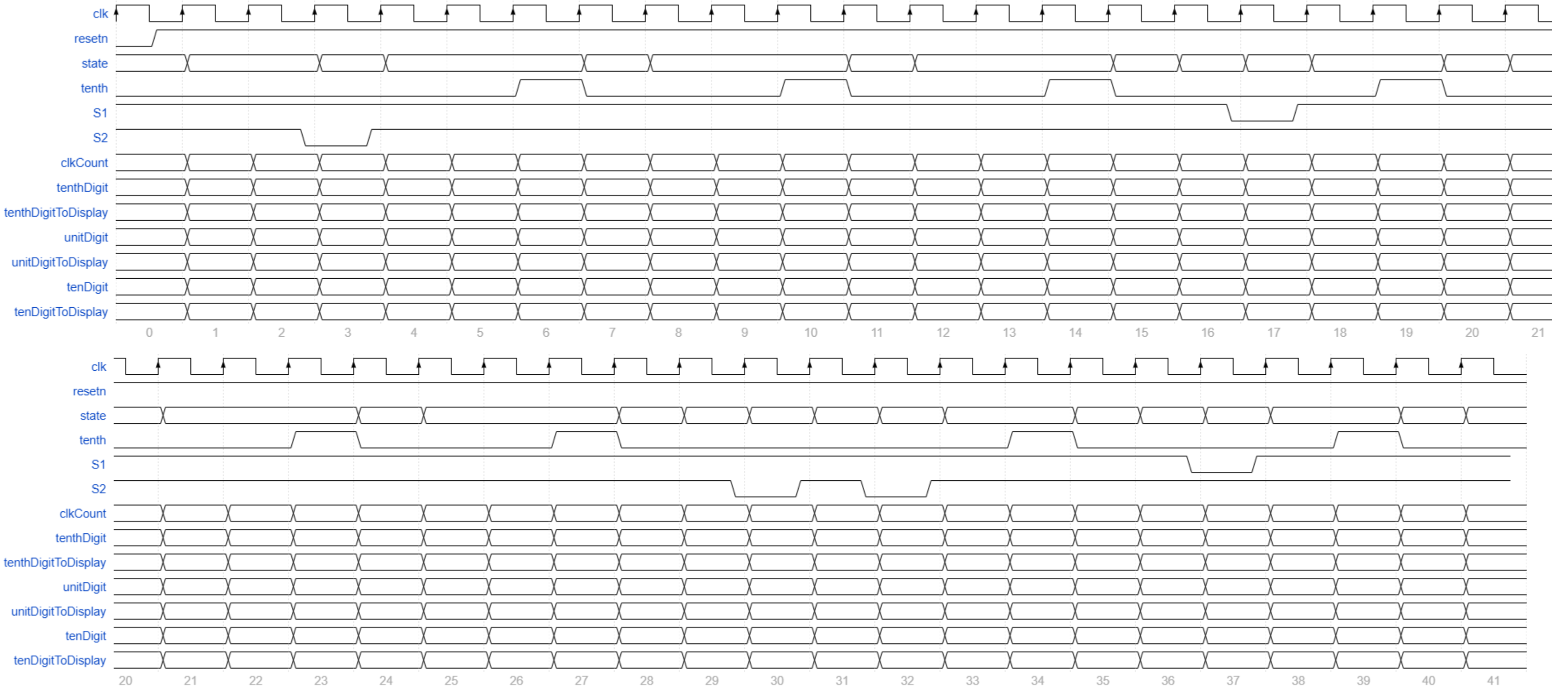

Figure 1.4: Timing diagram that you will run on the testbench.

The goal of the testbench was to cover every transition arc in the state diagram. This goal was not achieved; several transition arc was not taken in the testbench, which one was it? I'd suggest double checking this transition during the testing of the stopwatch when it is downloaded onto the Cyclone V GX development board.

Run the testbench using the provided do file and compare the output against Figure 1.4 You will probably need to modify the do file to make it work with your design. Address any problems in the stopwatch before proceeding.

## 1.5 Stopwatch Synthesis

When you created the datapath, you intentionally designed the timer counter to count up from 0 to 2 in order to expedite execution of the simulations. Before you synthesize the datapath, you need to undo this. I would suggest leaving the relevant constants in your code and just comment them out. Immediately following each commented constant, put the constant that you need for the datapath to operate correctly on the Cyclone V GX board. I've summarized these changes in Listing 1.1.

Listing 1.1: Changes to the datapath that will allow it to run properly on the Cyclone V GX development board.

```
// parameter N = 4;
parameter N = 24;

// localparam    tenthSecondConstant    = 4'h000002;
localparam    tenthSecondConstant    = 24'h4c4b40;

//localparam    zero24  = 4'h000000;
localparam    zero24  = 24'h000000;
```

While you are at it, make sure that you remove the testbench as the top-level module and make the stopwatch the top-level module. Then run the analysis and elaboration tool to make sure that the changes in Listing 1.1 did not create any warnings or errors

The next step is to create the mapping of stopwatch module inputs and outputs to the pins of the FPGA and by extension the input and output devices on the Cyclone V GX board. Use the inputs and outputs shown in Figure 1.2 and the information in the FPGA development board User Guide to complete the following pin assignment tables.

| S2 | Key[3] | Y16 |
|---|---|---|
| S1 | Key[2] | |
| resetn | Key[0] | |
| clk | CLOCK_50 | R20 |

| Segment | tenHex Hex2 | unitHex Hex1 | tenthHex Hex0 |
|---|---|---|---|
| seg[6] | | | |
| seg[5] | | | |
| seg[4] | V20 | | |
| seg[3] | | | |

| Segment | tenHex Hex2 | unitHex Hex1 | tenthHex Hex0 |
|---|---|---|---|
| seg[2] | | | V17 |
| seg[1] | | | |
| seg[0] | | AA18 | |

After making the pin assignment, download and test your design.

## 1.6 Turn in

You may work in teams of at most two. Make a record of your response to the items below and turn them in a single copy as your team's solution on Canvas using the instructions posted there. Include the names of both team members at the top of your solutions. Use complete English sentences to introduce what each of the following listed items (below) is and how it was derived. In addition to this submission, you will be expected to demonstrate your circuit at the beginning of your lab section next week.

### Stopwatch Simulation

- Completed Figure 1.4. Please:
  - Paste the images in landscape format into your solutions
- Complete simulation. Please:
  - Screen shot the simulation into three parts,
    - from 0 to 400ps
    - from 400ps to 800ps
    - from 800ps to 1200ps
  - Paste the simulation in landscape format into your solutions

### Stopwatch Synthesis

- Completed pin assignment table
- Demo working stopwatch