Digital Laboratory Lab04 Report

(Experiment Record Template)

Watch Lab video before the lab day. Playlist:

**You can finish the lab at home before you enter the classroom.**

Note:

Every Vcc voltage of 74 series ICs should be set to 5 Volts.

You don’t need consider where the sequence will start. TAs check only stable state after circuit operating for a while.

**Experiment-01**

**Design a sequence generator which is satisfied the following requirements:**

1. The set of sequence contains 16 elements from 0 to 15.
2. Show these numbers on the 7-segment display in particular form shown below.

 ([source](https://en.wikipedia.org/wiki/Seven-segment_display))

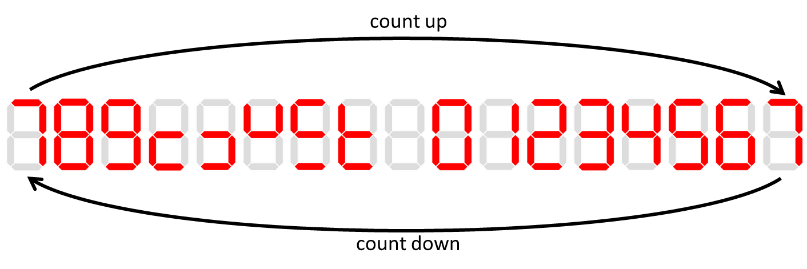
1. The clock source of this system is produced by 555 timer IC.
2. Counting direction will change when reaching boundary number.
3. Sequence generating rate and counting boundary are assigned corresponding to your ID number.

|  |  |  |
| --- | --- | --- |
| Unit digit of your ID | Counting boundary number | Sequence generating rate |
| **0** | 1 | ≈ 2 Hz |
| **1** | 3 | ≈ 4 Hz |
| **2** | 2 | ≈ 2 Hz |
| **3** | 5 | ≈ 4 Hz |
| **4** | 6 | ≈ 2 Hz |
| **5** | 7 | ≈ 4 Hz |
| **6** | 8 | ≈ 2 Hz |
| **7** | 9 | ≈ 4 Hz |
| **8** | 4 | ≈ 2 Hz |
| **9** | 0 | ≈ 4 Hz |

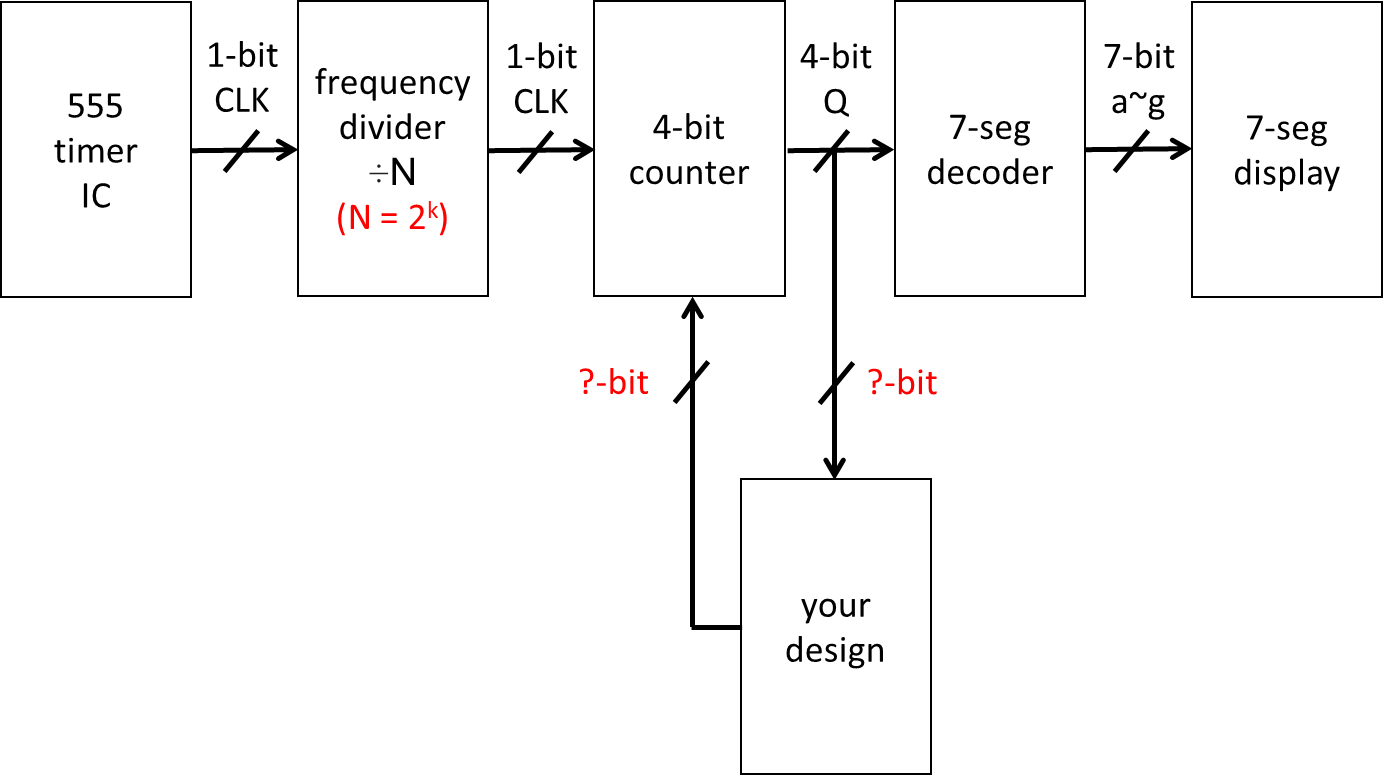
**Example**

ID = 6712345 🡪 unit digit = 5

🡪 change counting direction when reaching “7” and count a number per 1/4 second



**System Architecture reference**



**555 Timer IC astable oscillator circuit diagram**

|  |  |
| --- | --- |
| ([source](https://www.digikey.tw/en/resources/conversion-calculators/conversion-calculator-555-timer)) | Note: theoretical oscillation frequency = 16 Hz at pin3; OUTPUT. |

**Pinout**

|  |  |  |
| --- | --- | --- |
| electrolytic capacitor | common-anode 7-segment display | |
| ([source](https://www.yamanelectronics.com/capacitor-anode-cathode-identification/)) | Seven segment display interfacing with 8051(89c51,89c52) Microcontroller  ([source](https://www.engineersgarage.com/seven-segment-display-with-8051/)) | CA display, segment pins b and c set to LOW.  ([source](https://www.jameco.com/Jameco/workshop/TechTip/working-with-seven-segment-displays.html)) Note: Short-circuit protection resistors are all 1k Ohm. |

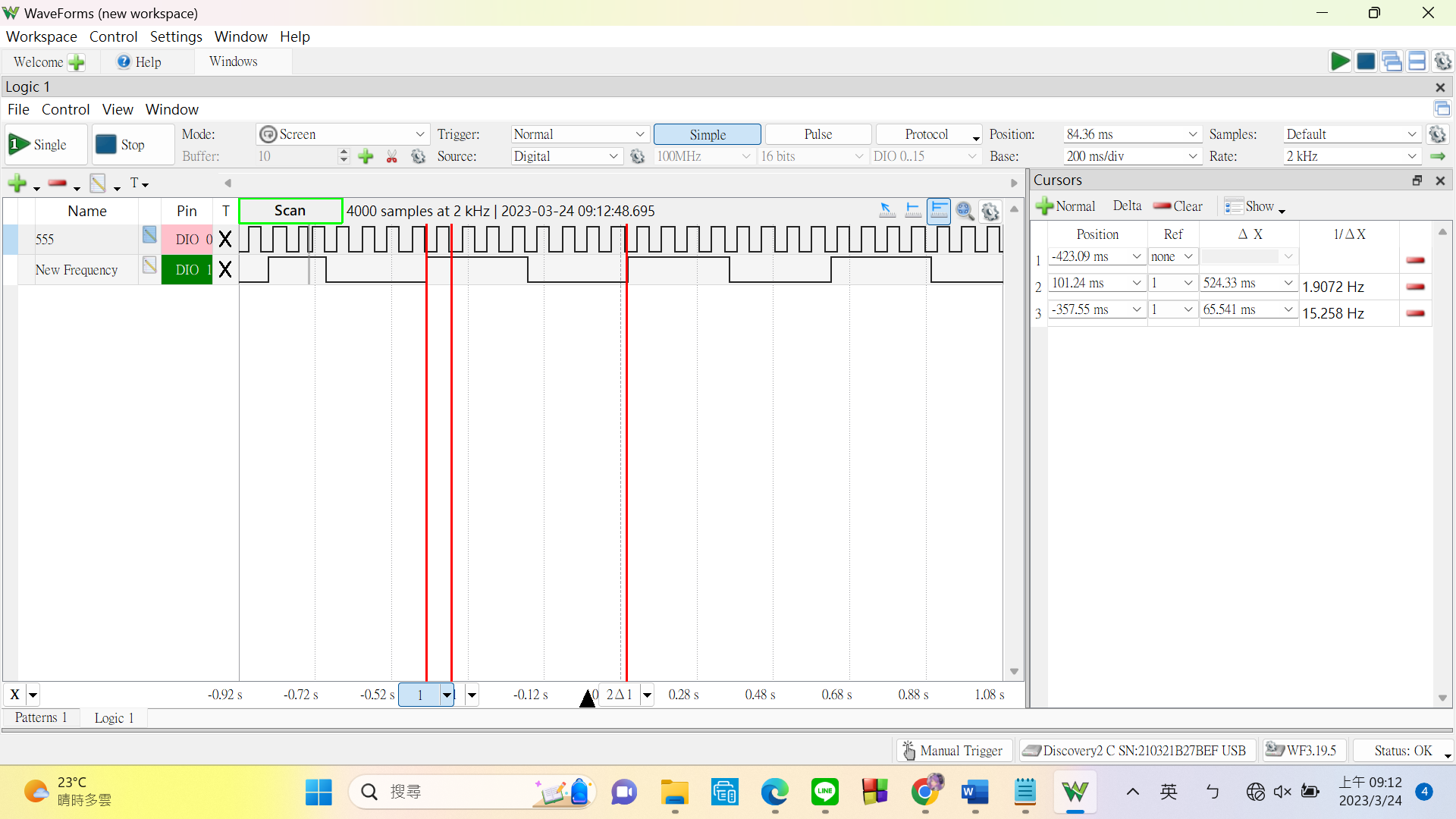
Live demonstrate this system to the TA.

Your sequence generator demo video link:

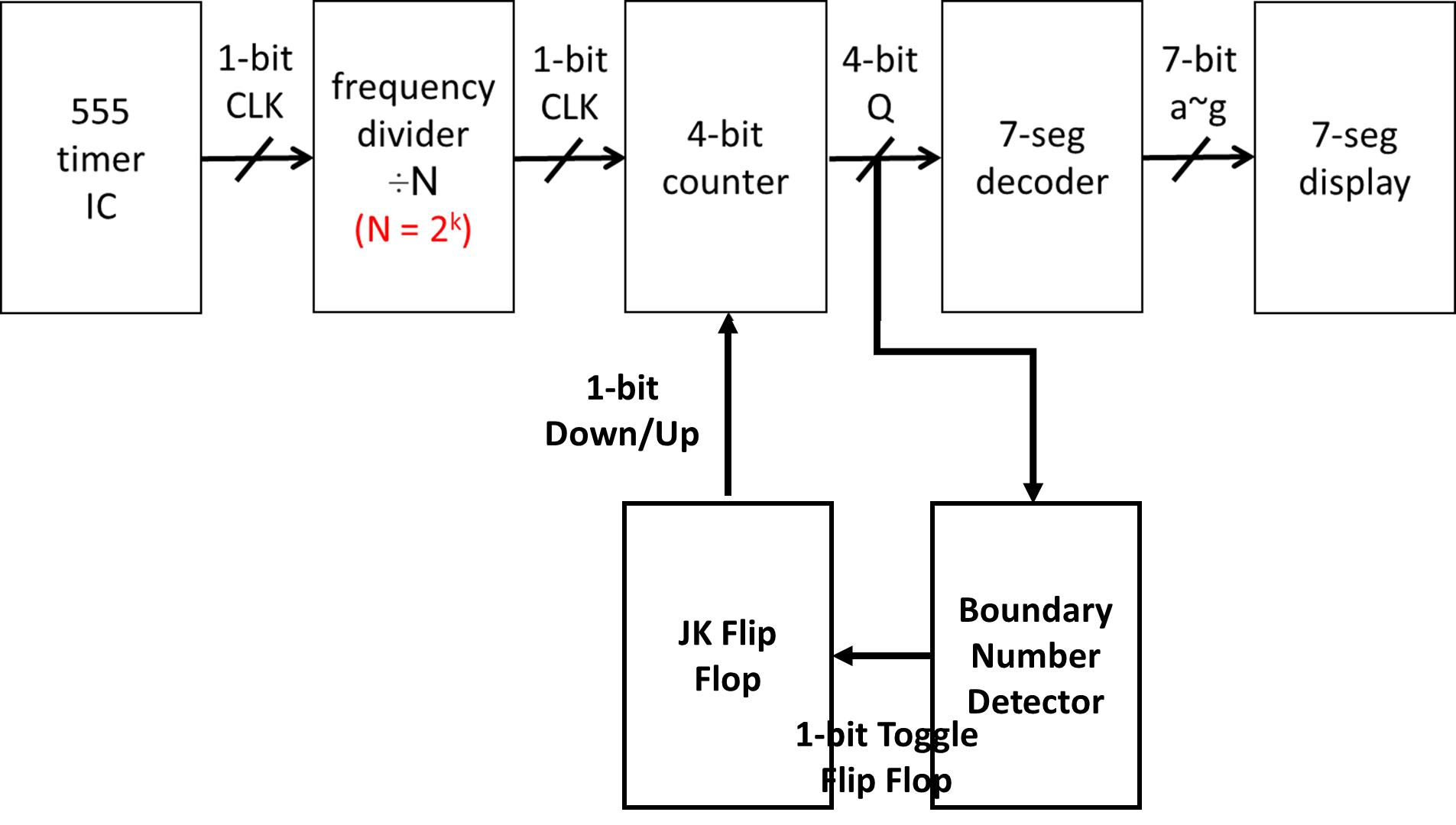
(Attach your video URL. DO NOT INSERT VIDEO FILE INSIDE THE REPORT)

<https://www.youtube.com/shorts/9yoYv6iPLZw>

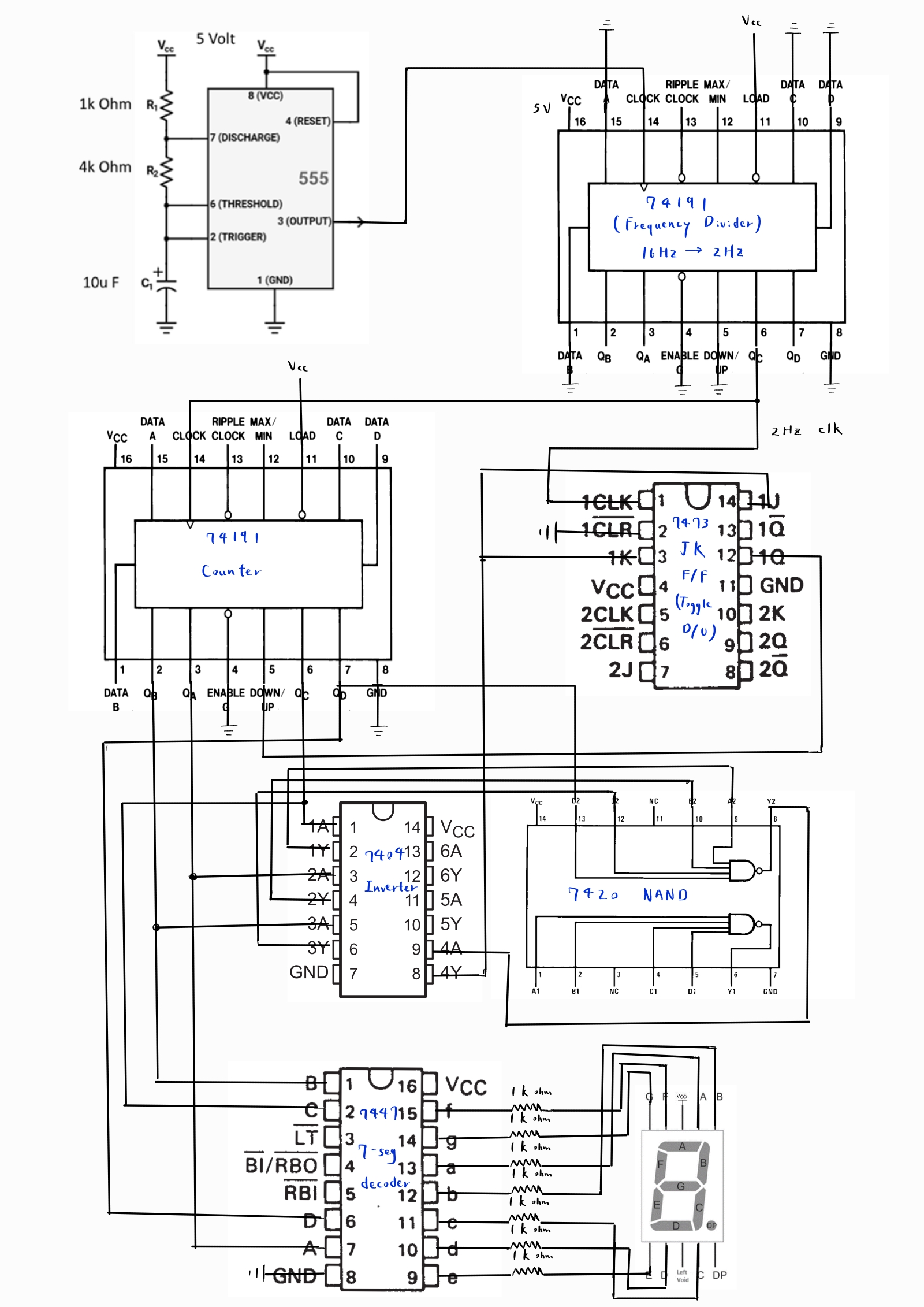
Generating rate measurement (waveform, parameters, etc.)



System Architecture of your design



Circuit diagram (connections of all resistors, capacitors and ICs)



Questions: **(Solve it at home. No live demo)**

**(Ask your own questions and answer them by yourself)**

**Last time, we used ceramic capacitors. Why are we swapping to electrolytic capacitors this time**

After some research, here’s what I found:

Basically, electrolytic capacitors and ceramic capacitors have different characteristics that make them better suited for different applications.

Electrolytic capacitors have a high capacitance per unit volume and are ideal for applications that require large capacitance values. They are also polarized, which means they have a positive and negative terminal, and must be connected with the correct polarity to function properly. Electrolytic capacitors are commonly used in applications that require large capacitance values.

Ceramic capacitors, on the other hand, have a lower capacitance per unit volume but can operate at higher frequencies than electrolytic capacitors. They are also non-polarized, which means they can be connected in either direction. Ceramic capacitors are commonly used in high-frequency circuits.

Therefore, the choice between electrolytic and ceramic capacitors depends on the specific requirements of the circuit. If you need a high capacitance value for a low-frequency application, an electrolytic capacitor would be a better choice. If you need a capacitor for a high-frequency application, a ceramic capacitor would be a better choice.



**This week will be the end of working with breadboards and physical circuits. What’s my main takeaway?**

Debugging circuits is more if not just as difficult, annoying, and time-consuming as debugging C++ code. However, I did learn a lot more about circuits in general through this kind of hands-on practice. I have understood:

1. How to read circuit diagrams.
2. How to fix my circuit(This usually involves removing everything from the board and reconnecting them……)
3. How to neatly connect wires(Originally, I paid no mind to how my wires looked on the board. However, messy wires make it extremely hard to debug. When I saw that someone else managed to keep their board nice and clean, I decided to follow suit. This is evident in how my Lab 4 looks. The wire connections look much cleaner and easy to follow.)
4. Laid the groundwork for Verilog(Since we are moving over to learning Verilog, it’s important to know the basics of circuits. I find that my previous practices were helpful when I was working on Lab 5)
5. Understanding what we learned in last semester’s Logic Design. (After learning about flip flops last semester, we didn’t really have a chance to use them. I like that this course gives us an opportunity to utilize what we learned. This allowed me to review and better understand what was taught in Logic Design. )