

Creating a Communications and Remote Control Environment (Labview 1)

Sejin Jeon
Sogang University Physics Department
Student ID 20231262

(10th Week Post-Experiment Lab Report)

I. EXAMPLE 1 (DATA & ANALYSIS)

For the first example, a simple VI (visual interpreter) was created. The process involved creating a new VI file named “plus and minus.vi”, creating 2 numeric controls, and creating 2 indicators that would interact with each other in a specific way. Overall, the “A” numeric would add with the “B” numeric to ultimately indicate the result of the operation of the “C” indicator, whereas the “D” indicator would indicate the difference. For details, refer to the image below.

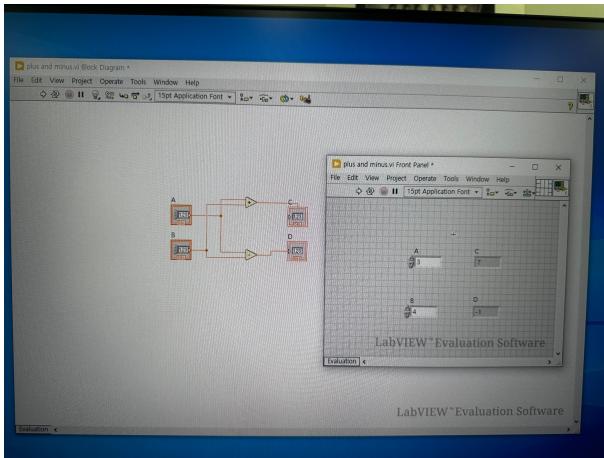


FIG. 1

II. EXAMPLE 2 (DATA & ANALYSIS)

In the second example, a front panel was constructed, where the slide object would change the indication of the gauge object. On the other hand, three boolean controls were selected to be connected in a particular manner. The first control switch, which would be controlled manually, would indicate two (boolean) values, on and off, and the third indicator switch would show the opposite value due to the “not” node, whereas the second indicator switch would indicate/show the same (boolean) value. For details, refer to the images below.

III. EXAMPLE 3 (DATA & ANALYSIS)

In the third example, the while structure was used to create a new VI named “random number generation.vi”.

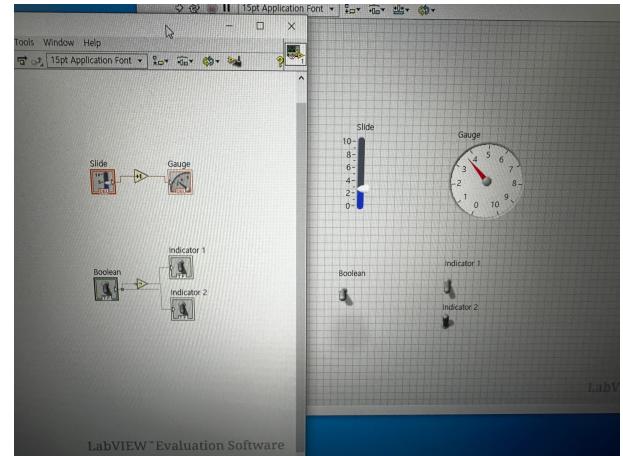


FIG. 2

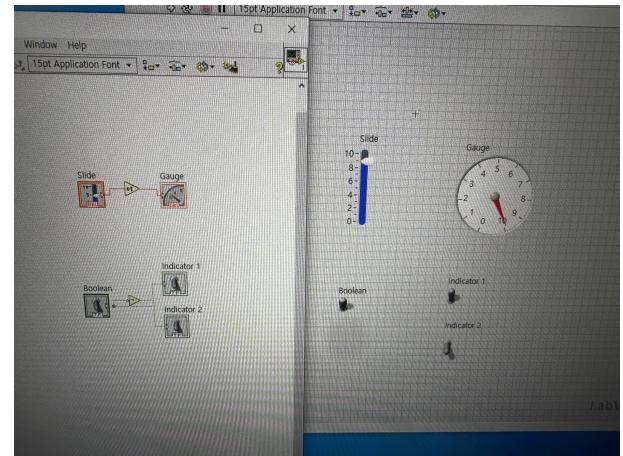


FIG. 3

The code ultimately created a random number and chart made of random numbers using a numeric indicator, waveform chart, and a stop object. After the code was ran, the stop button would urge the random number generation to be stopped, and the time ran before the stop button was pressed would be indicated on the numeric indicator. For details, refer to the image below.

IV. EXAMPLE 4 (DATA & ANALYSIS)

For the fourth example, a variation of the third example was created, where a code was created that would stop

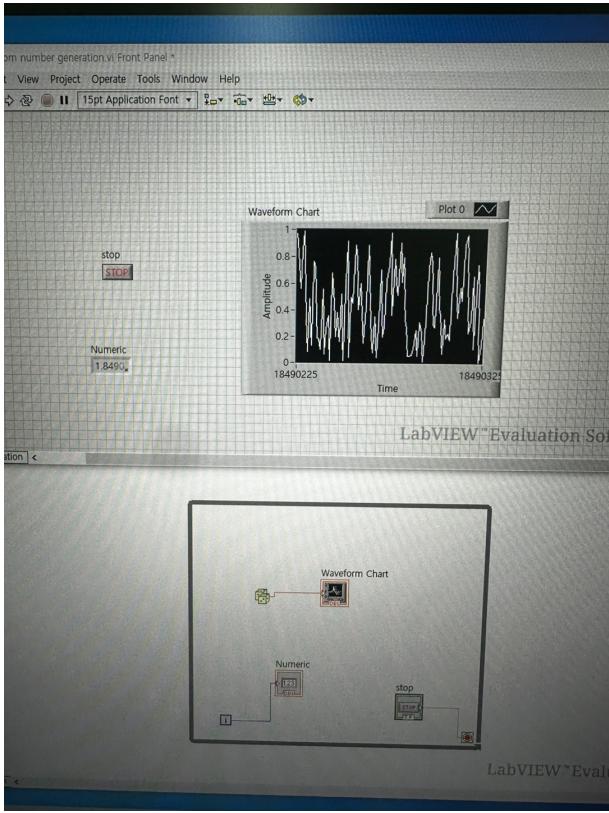


FIG. 4

if the random number generator would create a number identical to the number inserted in the control numeric. The code structure was follows. First, one control and one indicator numeric would be placed in the front panel, and the numeric control's properties would be changed such that it's range of values would fit our purpose. In a while argument, the random generator function would be multiplied and rounded up tot eh nearest whole number, and connected to the numeric indicator. This value would be constantly compared with the indicator control out of the while structure. For details, refer to the image below.

V. EXAMPLE 5 (DATA & ANALYSIS)

For the fifth example, a noise containing signal would be filtered through a running average structure. The code structure would be as follows. A while loop would be ran, and inside the while loop, the time information would become the horizontal axes of the sine function that would indicated through the waveform chart. With this original function placed, we would add noise by simply connecting the output sine function with a random number generator that would add small incremental values to the output waveform chart.

The output noise containing function would be analyzed through a 5 point running average filter, created through

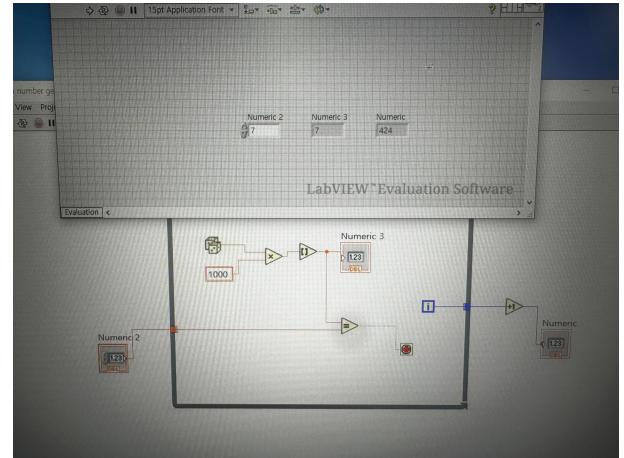


FIG. 5

connecting the values within the loop through a shift register to a compound arithmetic operator, dividing the five different values throughout time to indicate the final output on the waveform chart. For details, refer to the figures below.

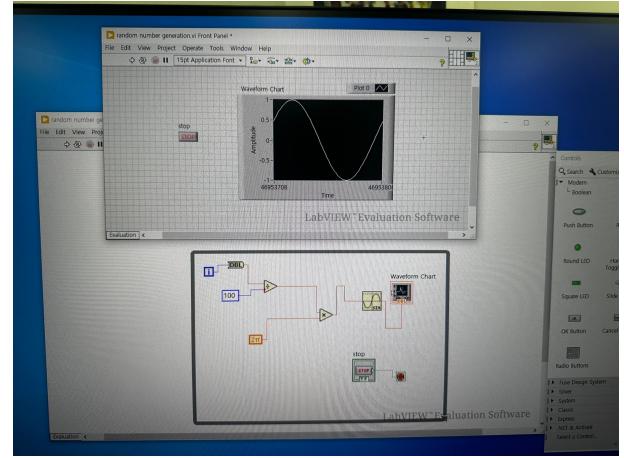


FIG. 6

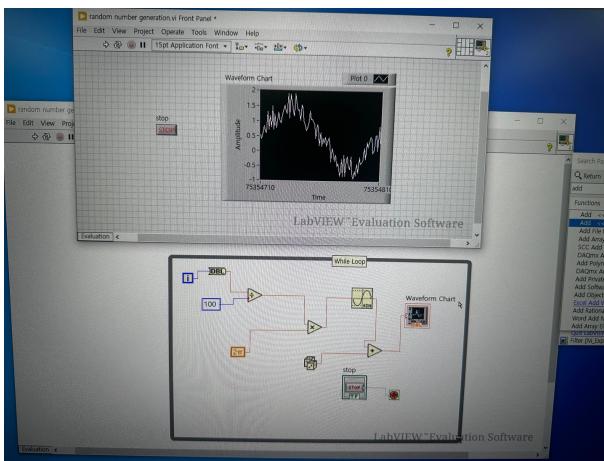


FIG. 7

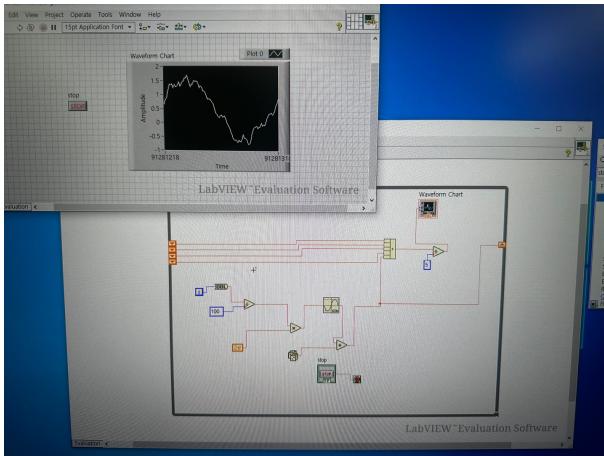


FIG. 8

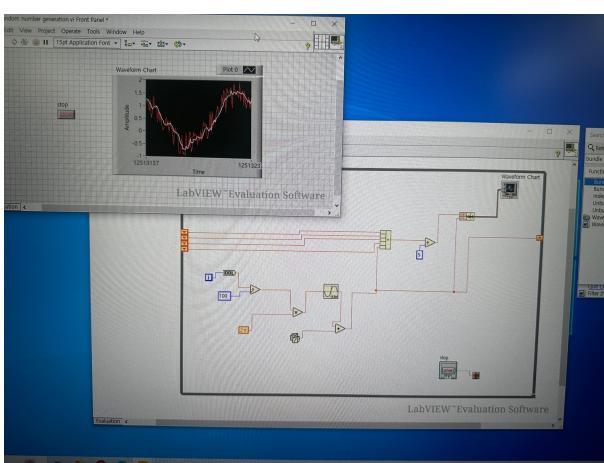


FIG. 9

VI. EXAMPLE 6 (DATA & ANALYSIS)

For the sixth example, different wave form data were plotted on a graph. First, a new VI was created named “waveform generation.vi”, and an enumerate element was created. Different cases were then created (sine, square, saw, triangle, and noise waveforms), and corresponding waveform graphs were created as seen below.

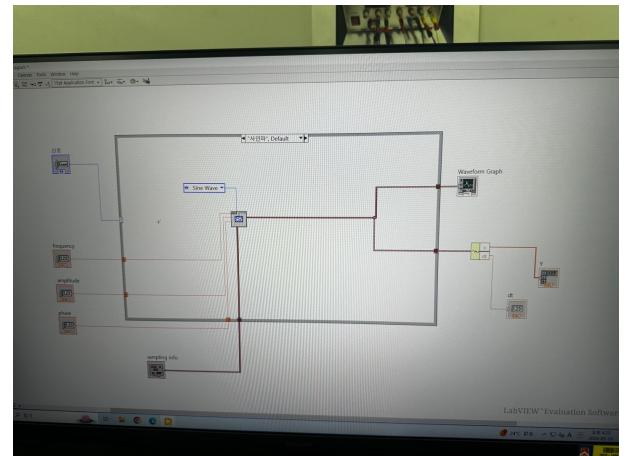


FIG. 10

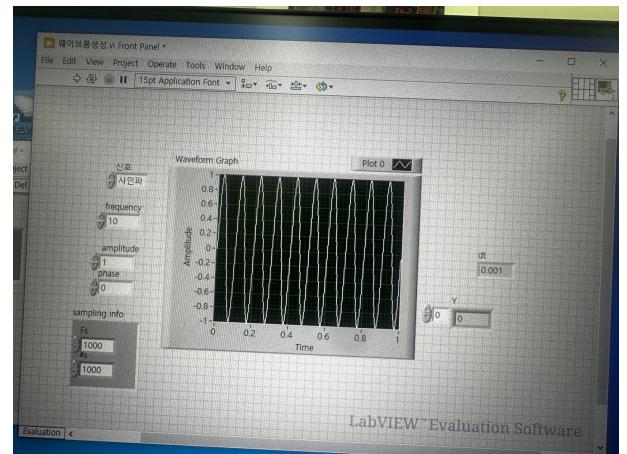


FIG. 11

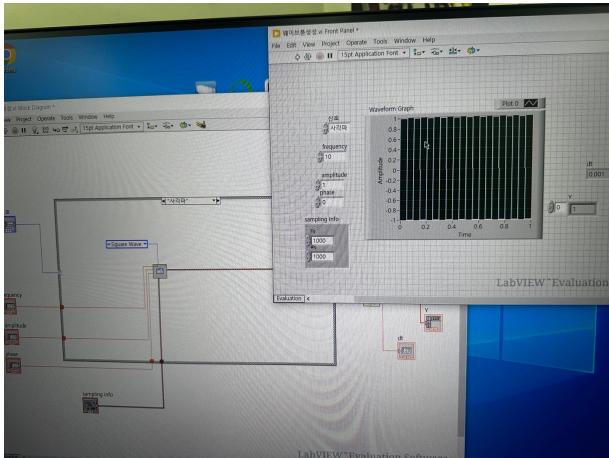


FIG. 12

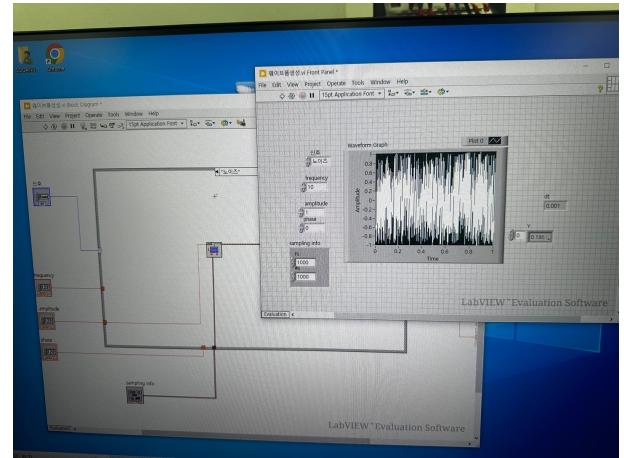


FIG. 15

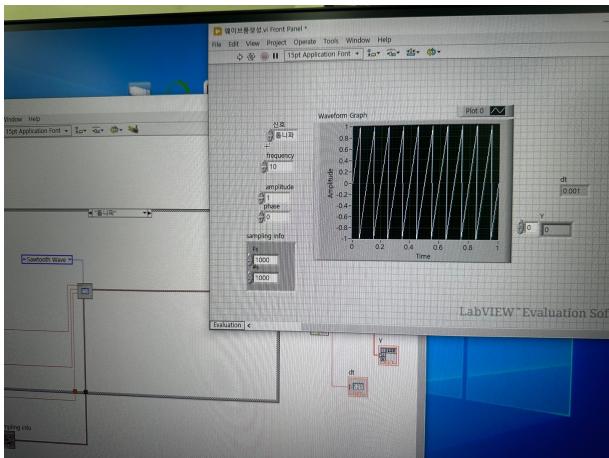


FIG. 13

VII. EXAMPLE 7 (DATA & ANALYSIS)

For the final example, the 7th example of this experiment, a text file saver was created, where it would create a file with certain information and save it in a predestined file. Refer to the figures below for details.

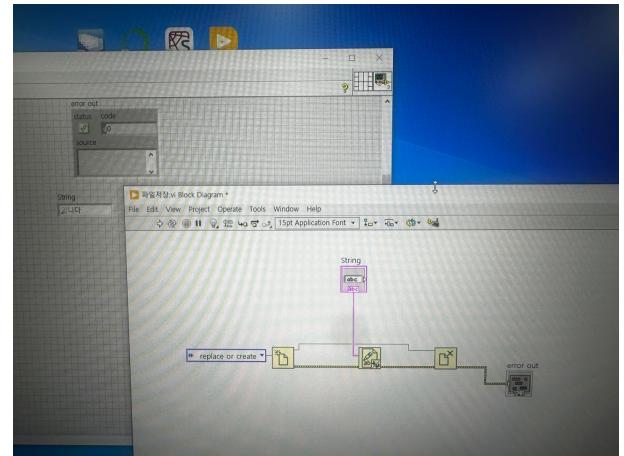


FIG. 16

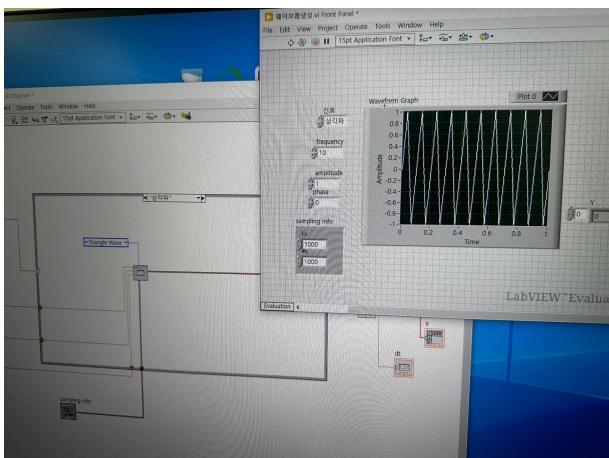


FIG. 14

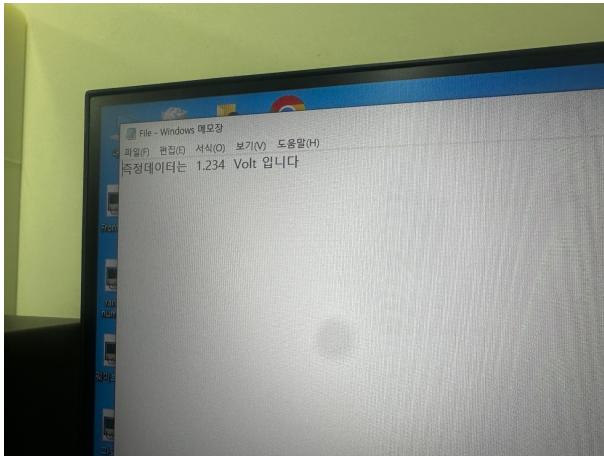


FIG. 17

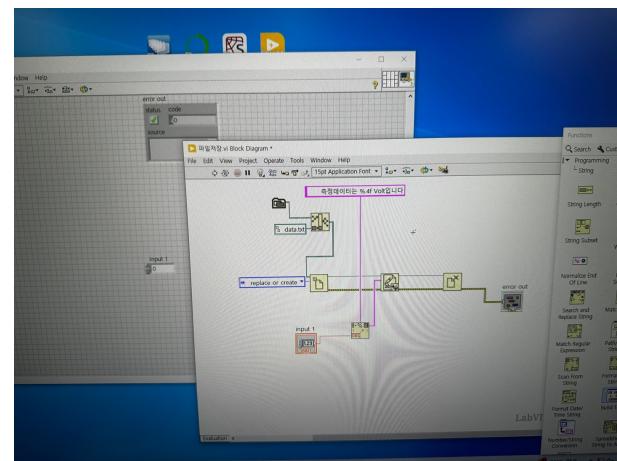


FIG. 20

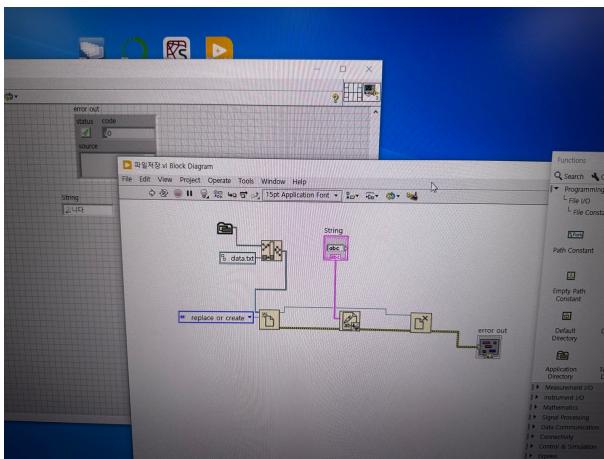


FIG. 18

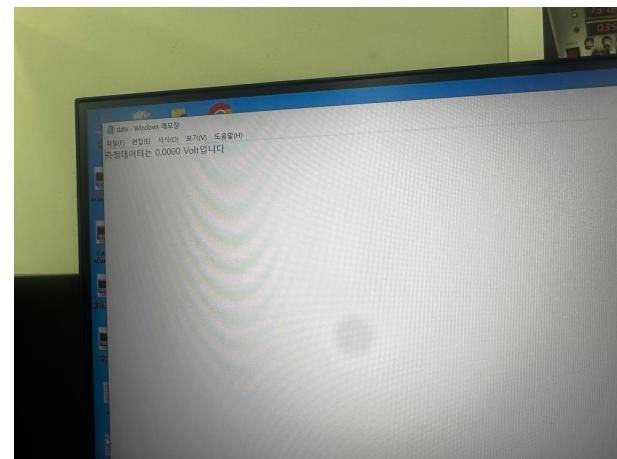


FIG. 21

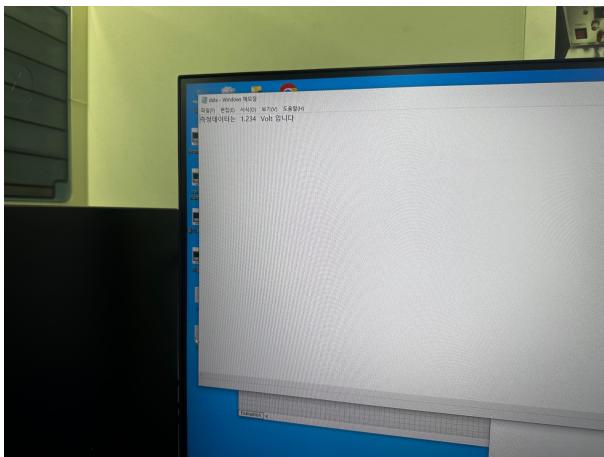


FIG. 19

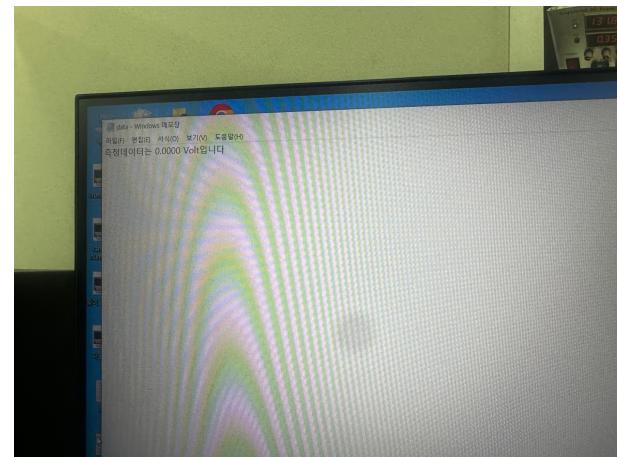


FIG. 22

VIII. DISCUSSION

Throughout the experiment there were a few errors that needed to be evaluated and there were also quite a few improvements that could be made. They are listed as paragraphs below.

Alignment of the elements made the diagrams and templates seen more readable and easily interpretable,

and if these were done for the while structures, it would have made the overall code much more easily understandable.

Comments were not made throughout the layout, contrary to the Python codes that were created previously. If more comments were made, it would have been better and made the whole layout more readable.

[1] THE SOGANG UNIVERSITY PHYSICS DEPARTMENT. Experi-

mental physics 1 manual. “*Creating a Communications and Remote Control Environment (Labview 1)*”.