Süleyman Emre CAN - 2093524
Yekta Demirci - 2093607
31.05.2019

EE 497 Real-time Applications of Digital Signal Processing
PROJECT REPORT

Introduction

In this project, we implemented overlap and save method firstly in MATLAB and then in C which runs in myRIO. On the MATLAB part, we implemented the overlap and save algorithm by using fft. After being sure that the algorithm returns correct results, code is implemented in C, converted into .so file, and placed in the myRIO to run.

Overlap and Save Method

Since we are dealing with real time implementations and large data to process signals, it is usually required that the algorithms should not spend too much time in one of the operations, such as linear convolution operation. To shorten the amount of time spent in convolution operation, overlap and save method is highly beneficial. During the overlap and save method;

- P-1 amount of zeros are padded at the beginning of the M length data x[n].
- L point circular convolution of data and P length filter, h[n], is obtained in windows of length L on the begining data through fft and ifft operations.
- First P-1 samples of the L point circular convolution is deleted because they are not useful.
- Window is shifted by L-(P-1) points
- Circular convolution is obtained in the new window, non-useful part is deleted.
- Correct part of the convolution is concatenated in the result array.
- Window is shifted and the deletion process is continued in a loop until the end of the data. Since the end of the data might not be involved in the final loop due to its length, there are also zeros are padded at the end, and deleted while obtaining final result.
- M+P-1 length of final result array is obtained through loops, which corresponds linear convolution of data x[n] and filter h[n]

is implemented in our code.

Visualisation of the process can be seen in figure.1

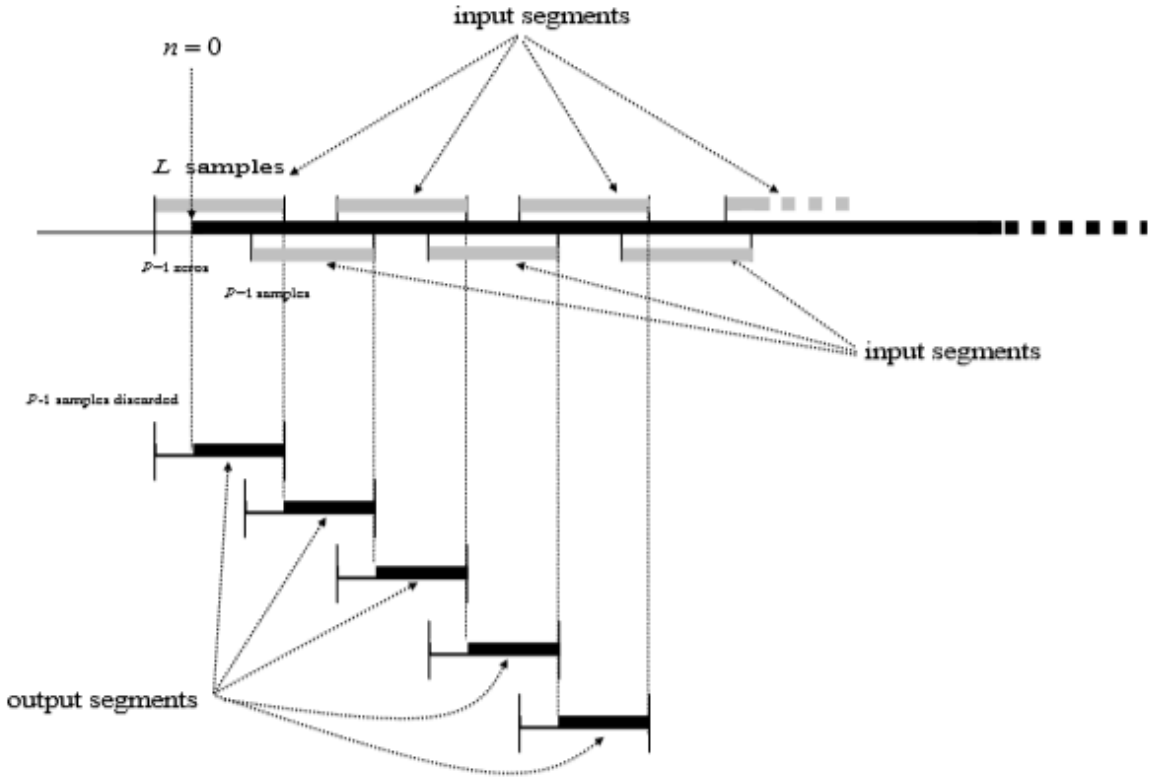

Figure 1: Overlap and Save visualization

Implementations and Results

MATLAB Implementation

On the MATLAB, exact same algorithm, which given in introduction part is implemented with randomly generated data, randomly generated filter, and it is compared with the linear convolution operation. All the required fft, ifft, and linear convolution operations are done by built-in functions of MATLAB. Results of the comparison can be seen in figure 2.

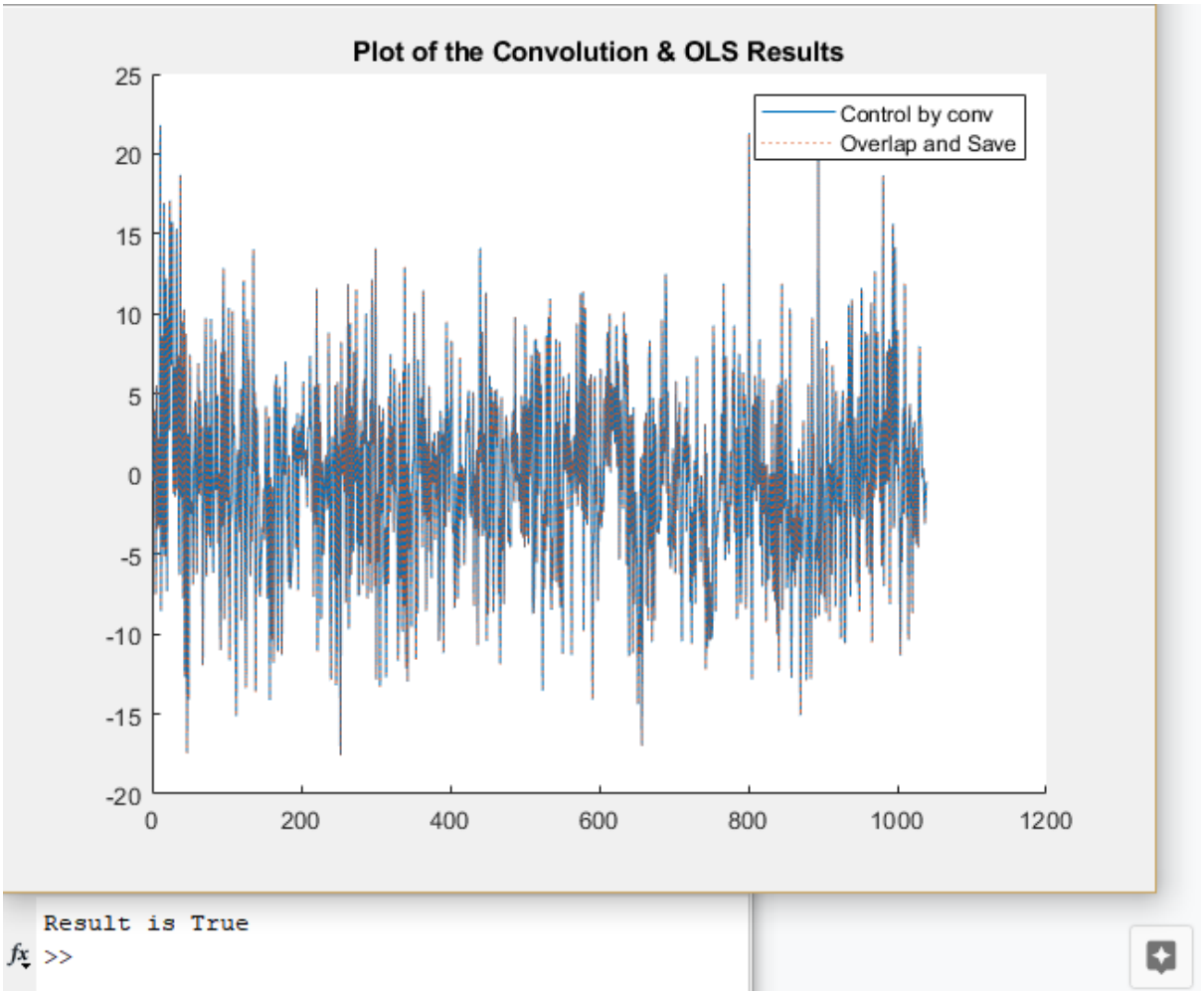

Figure 2: Overlap and Save method results on the MATLAB plotted over linear convolution results for control

C Implementation

After being sure that the algorithm works correctly, exact same algorithm is implemented in C. Since the complex calculations are problematic on C, a DFT operation module is written to calculate the DFT of the window which returns the real & imaginary parts of the result in two different arrays. After the multiplication operation is done with the resulted arrays, similar process is done for IDFT operation and the real & imaginary parts of the circular convolution is obtained in two different arrays. After the required amount of loops, same results with MATLAB implementation is obtained. Through the code, everything is coded dynamically so input sizes are flexible. Since no library is used except math, the code worked smoothly and without any additional installation on myRIO. On the Eclipse, compilation and the transform of the C code to .so file is done and placed in myRIO via SSH to run.

LabView block diagram and test results can be seen in figure 3 and figure 4.

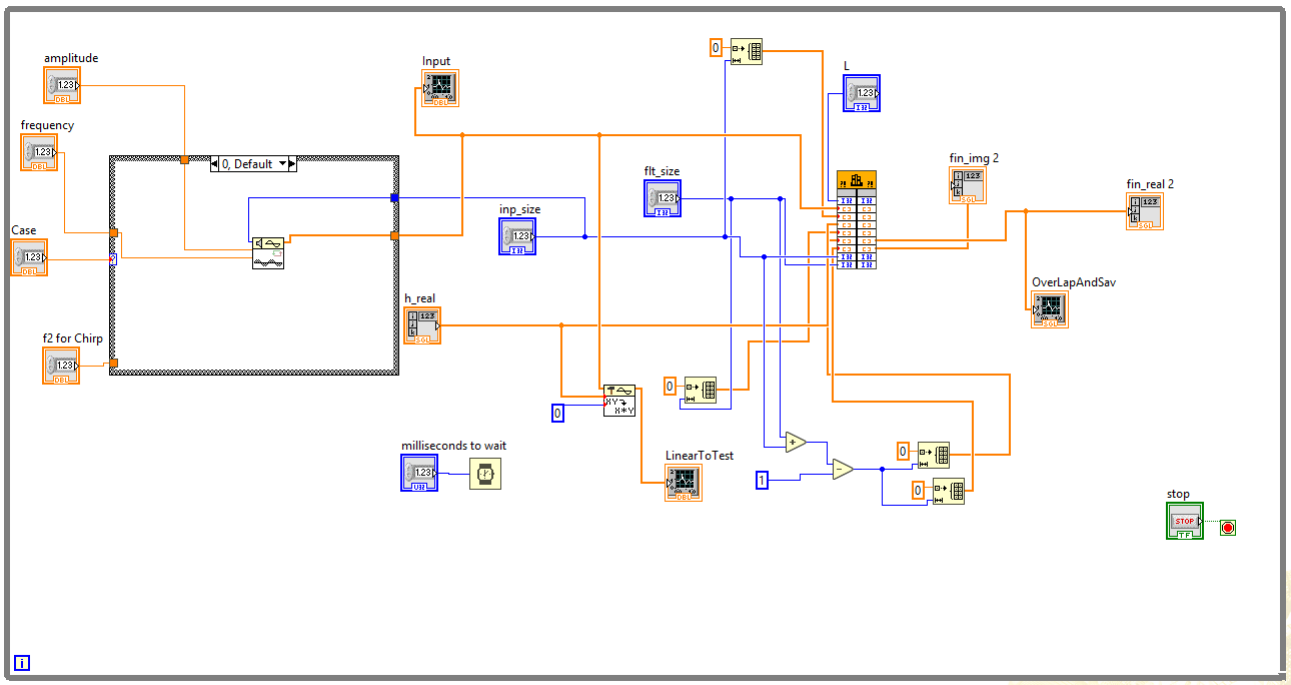

Figure 3: LabView Block Diagram of Implementation

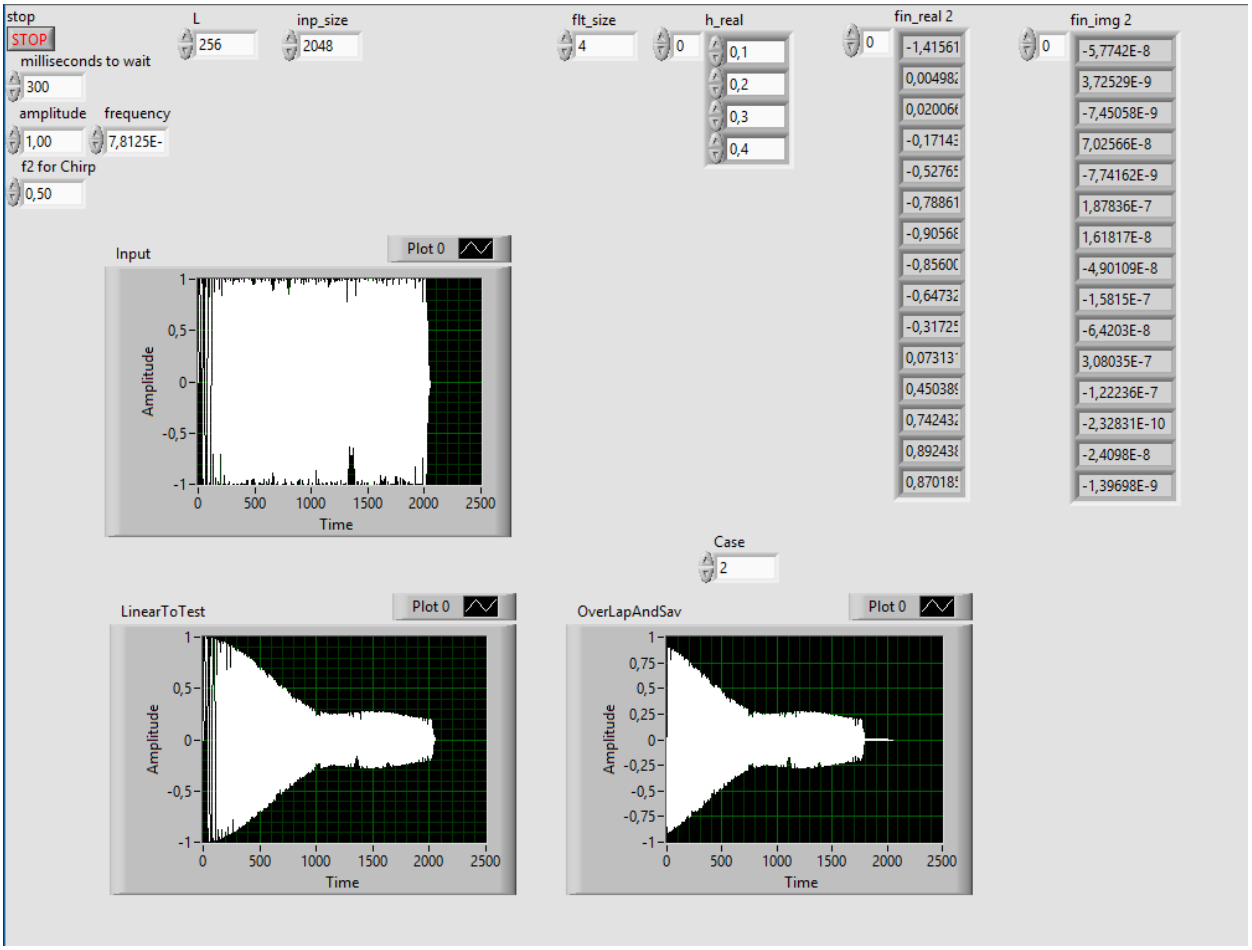

Figure 4: Overlap and Save result compared with linear convolution

Conclusion

During this project, we implemented overlap and save method in two different environment, MATLAB and C. We also learned how user defined C functions can be run in myRIO. This was a great project to observe how can we overcome the time requirement of the real-time implementations, and to have experience on the process of running user defined functions in myRIO environment, which significantly reduces the dependency on the built-in libraries.