Volume 5, Issue 10, pp. 8-19, 2021.

International Journal of Scientific Engineering and Science
ISSN (Online): 2456-7361

Design and Optimization of Fuzzy-based Digital FIR Kernel Filter for Real-Time Control Applications

Kelvin N. Nnamani[1*], Stephen U. Ufoaroh[2], Scholastica U. Nnebe[3]
[1, 2, 3]Department of Electronic and Computer Engineering, Nnamdi Azikiwe University, Awka, Anambra State, Nigeria
Email address: kn.nnamani(at)stu.unizik.edu.ng

Abstract— *21st century farmers are prone to the dangers of climatic changes resulting from intense temperature and relative humidity in their nursery beds of fruits and vegetables. For optimum growth of those fruits and vegetables without having to visit their farms often, digital FIR filters that control the Low pass frequency threshold at 0.3π rad/time span , blocking the high pass frequency signal at 0.7 π rad/time span as well as control the high pass frequency threshold at 0.7π rad/time span , blocking the low pass frequency signal at 0.3 π rad/time span were designed, employing optimization methods such as Least square method, Parks McClellan Algorithm and window methods setting the filter length of 200 samples; galvanized with convolution techniques. It is interesting to know that the digital FIR filters with built-in temperature and humidity sensors control fuzzy filtering (using Gaussian membership function with 6 base rules) in the greenhouse nursery bed; hence the name Fuzzy Digital FIR kernel Filter for Real-time control Applications. The temperature and relative humidity of the next-generation greenhouse been controlled are in the range of 15-30℃ and 50-90% respectively. All through the design, MATLAB fuzzy logic, DSP, Filter tool boxes and MATLAB File editor (M-File) were utilized for the simulation, modeling and generation of C-programming codes.*

Keywords— *FIR Filter, MATLAB, Greenhouse, Low pass, high pass, Convolution, Fuzzy logic, Temperature, Humidity.*

I. INTRODUCTION

The idea of Fuzzy modeling and control is deployed in the design of Digital FIR Filter for Low pass and high pass Filtering of a stipulated threshold of frequency passband while the unrequired threshold is blocked. Fuzzy control is a type of control that makes use of fuzzy set theory. Fuzzy control has two distinct advantages. Fuzzy control, for starters, provides a novel framework for executing control principles that are frequently dependent on knowledge or linguistic descriptions. Second, fuzzy control [7] provides an alternative paradigm for facilitating the design of non-linear controllers for plants that rely on generally uncertain control, which is difficult to link to traditional non-linear control theory. The possibility of improving the accuracy of climate control in next generation greenhouse cultivation by way of model based filtering [1],where temperature and humidity is taken cognizant of for automatic real-time control of nursery cultivated with fruits and vegetables, it is ideal to set up a required threshold for frequency responses when average daily temperature and humidity are set below or above normal to control its filtering action for optimum growth of plants in the nursery bed by use of sensors(temperature and humidity sensors) aimed at filtering out a high frequency noise when a low frequency signal is passed and a low frequency noise when a high frequency signal is passed without having to visit the nursery often. This can be done by installing greenhouse controllers at specified time interval [2], hence the need to design a fuzzy digital filter for the control. Common filtering objectives are to improve the quality of a signal, to extract information from signal or to separate two or more signals previously combined [8].

The digital filter is a discrete system, and it can do a series of mathematic processing to the input signal, and therefore obtain the desired information from the input signal [6]. Digital filter is a linear time invariant system (LTI) which does not vary with time. Digital filters have high accuracy, easy to simulate and design, flexible than analog filters [3]

II. REVIEW OF RELATED WORKS

Two scenarios indicative of modern greenhouse practice were used to test the filter performance [1] in terms of monitoring accuracy of average temperature and humidity inside a greenhouse compartment. Temperature and absolute humidity were measured with a sensor grid inside a big and small greenhouse compartment with a 5-min sample resolution. Sensor grids that were spatially scattered were used in both circumstances. The instances were recreated by using a single sensor's readout as the filtered signal. The reference data was obtained by taking the sensor grid's spatial average. The root mean squared error (RMSE) between the filtered data and the reference data was used to measure monitoring accuracy. The root mean squared errors (RMSEs) of the unfiltered signals (coming from single sensors) were 0.43°C and 0.48 gm^{-3} for greenhouse A and 0.80°C and 0.64 gm^{-3} for greenhouse B, respectively, when compared to the average over all sensors as reference signal. By altering the time step and the magnitude of state noise covariance, the impact of the filter settings was investigated. The robustness of these results in terms of reference signal accuracy was also verified by continually deleting one sensor from the grid. The significance of noise assumptions was checked by substituting manufactured white noise with the same covariance for the observed noise in state and output. By arbitrarily increasing the amounts of observed model and sensor errors, the relationship between model accuracy and monitoring accuracy was projected.

Temperature and Humidity of automatic-computer controlled greenhouse nursery bed was controlled by overhead ventilators, fan, heater and mister at every 10 seconds interval [2]. The generalized bell-shape (gbell) membership function was employed through the application of ‘If then rules’ and centroid method of defuzzification. In the design, the greenhouse was programmed whenever the fan or overhead ventilator are put on or close when the temperature goes above normal and the mister or heater is powered on when the humidity goes below normal and vice versa. Ten iterated base rules were formulated based on fuzzy decision with the parametric universe of discourse of range -30 to 95℃ for temperature and 5 to 80% of humidity with each constituting five degree of membership; though the analysis was based on feedback control using MATLAB fuzzy logic tool box.

A greenhouse was managed effectively using wireless network to control temperature, relative humidity, soil moisture and light intensity for 4-months period (between early winters and early spring), based on fuzzy logic system of the sensor-based data input. In the greenhouse, tomatoes was optimally selected at varied temperature range of 25℃ – 27℃ and the humidity value is in the range of 50% to 70% [9]. The condition of the tomatoes growth was monitored based on fuzzy control system optimization criterion using Zigbee protocol. From this it was deduced that the mean night values for December were estimated at 37 ℃ and the relative humidity at 83.1%. Soil humidity was 55.7% and the amount of light was 3512 lux; the corresponding calorific value was 4.58 kW, the cooling value was 5.42 μm, the irrigation value was 8.99 l, the shading value was 53.7 cm and the amount of light was 15400 lux; taking into account the heat loss gain of 7.01 kW. The mean daily value for December also accumulated 7.2 ℃ for temperature, 82.8% for relative humidity, 50.1% for soil humidity, and the amount of light was 11422 lux; the corresponding calorific value was 4.39 kW, the cooling value was 517 μm with an estimated heat loss of 6.62 kW.

There is a prediction of sinusoidal signal with time-varying frequencies using fuzzy filtering scheme based on finite Impulse Response (FIR) filter bank using interpolation of individual filter outputs [4]. There was derivation of the filter coefficient based on the sum of squared prediction error (SSPE) for filtering performance. Five Gaussian membership function with frequency range of 49-51 Hz was employed. The sinusoidal input signal with time-varying frequencies, x(n) was deployed at 1.67KHz, predicted at every period of 2 seconds and *If Then Rules* used to evaluate the frequency range of the filters as follow: If $f\ (n) < 49.625$Hz *Then* switch to filter designed at 49.5 Hz,

If 49.625Hz $<f\,(n) <$ 49.875Hz Then switch to filter designed at 49.75 Hz,

If 49.875Hz $< f\,(n)$ <50.125Hz Then switch to filter designed at 50 Hz,

If 50.125Hz $< f\,(n) <$ 50.375Hz Then switch to filter designed at 50.25 Hz,

If $f\,(n) \geq$ 50.375Hz Then switch to filter designed at 50.5 Hz.

A closed-form linear phase FIR filter design [5] concurrently possessing high efficiency and excellent transfer characteristic by employing Fourier spectrum of a convolution window was proposed. There was increase in the FIR filter order to 198 (i.e., filter length, L=199 and N=100) and specify passband cutoff frequency $w_p = 0.5\pi\ rad/s$, transition bandwidth 0.02π (accordingly, $M = 26$), $w_c(n)$ is derived by convolving an N-length Hamming window with an N-length rectangular window.

III. METHODOLOGY

There are two phases of this design methodology: Digital FIR Filter design, and fuzzy based modeling. First, in the design methodology of both Low pass and high pass Digital FIR Filter, the filter coefficients were generated using first principle and using filter design tool box and signal processing tool box. This is done by allowing a finite impulse response of length 3 units involving time span (n), input sequence, X (n), and output sequence y (n). The input has been assumed to be a casual sequence with the first non-zero sample occurring at n=0, and for calculating y (0) and y (1), set x (-1) =x (-2) =0.

The signal responses of impulse and frequency responses of low pass and high pass Digital FIR Filter were generated using park-McClellan Algorithm and other window techniques for Digital FIR Filter designs such as rectangular, Kaiser, hamming, and Han where filter design tool box, signal processing tool box and filter visualization and tool box were employed to achieve the same design for easy implementation of fuzzy decision.

3.1. Real-Time Digital FIR Filters Design

3.1.1. Design objectives

- To design a Low pass discrete FIR Filter that will allow a low frequency component and block a high frequency component signals Using MATLAB Software.
- To design a high pass discrete FIR Filter that will allow a high frequency component and block a low frequency component signals Using MATLAB Software.

3.1.2 Design Techniques for low pass and high pass Digital FIR Filter.

For simplicity, we assume the filter to be an FIR filter of length 3 with an impulse response:

h (0) =h (2) =β_0, h (1) =β_1

From the equation, we have

Y (n) =$\sum_{K=N_1}^{N_2} h(k)x(n-k), N_1 < N_2$. The digital filtering is performed using the difference equation:

$$y(n)=h(0)x(n)+h(1)x(n-1)+h(2)x(n-2) \quad (1)$$

$$y\ (n) = \beta_0 x(n) + \beta_1 x(n-1) + \beta_0 x(n-2) \quad (2)$$

Where y (n) & x (n) represent the output and input sequences respectively. Also, from our design, we chose the filter coefficients β_0 & β_1 such that the output of the filter becomes the cosine sequence with frequency 0.7π rad/time span. Now, using equation of the frequency response.

$$H[e^{j\omega}] = \sum_{K=N_1}^{N_2} h(k)e^{-j\omega k} \quad (3)$$

Which is seen to be a polynomial in$e^{-j\omega}$. The frequency response of the above filter is given by:

$$H[e^{j\omega}] = h(0)e^{-j\omega 0} + h(1)e^{-j\omega 1} + h(2)e^{-j\omega 2} \quad (4)$$

$$H[e^{j\omega}] = \beta_0 + \beta_1 e^{-j\omega} + \beta_0 e^{-j\omega 2}$$

$\text{Re} - \text{arranging, we have}$

$$H[e^{j\omega}]=\beta_0+\beta_0 e^{-j\omega 2} + \beta_1 e^{-j\omega}=\beta_0(1 + e^{-j\omega 2}) + \beta_1 e^{-j\omega} \quad (5)$$

$\text{But}, \cos\omega = \frac{1}{2}(e^{j\omega} + e^{-j\omega}),$

$$2\cos\omega = e^{j\omega} + e^{-j\omega} \quad (6)$$

$$H[e^{j\omega}] = \beta_0(e^{j\omega} + e^{-j\omega})e^{-j\omega} + \beta_1 e^{-j\omega} = \beta_0(2\cos\omega)e^{-j\omega} + \beta_1 e^{-j\omega} =$$

$$(2\beta_0\cos\omega + \beta_1)e^{-j\omega} \quad (7)$$

The magnitude & phase functions of the filter are

$$\left|H[e^{j\omega}]\right| = |2\beta_0\cos\omega + \beta_1|$$

$\theta(\omega) = -\omega$

Step I: In order to stop the low frequency component from appearing at the output of the filter, the magnitude function at ω_s =0.7 π rad/time span should be set to Zero. Similarly, to pass the low frequency component within any attenuation, we need to ensure to ensure that the magnitude function at ω_p =0.3 π rad/ time span set to 1.

Thus, the two conditions that must be satisfied are:

$$2\beta_0 \cos(0.7) + \beta_1 = 0 \qquad (8)$$

$$2\beta_0 \cos(0.3) + \beta_1 = 1 \qquad (9)$$

Or converting to degrees, we have

$$2\beta_0 \cos 40.1 + \beta_1 = 0 \qquad (10)$$

$$2\beta_0 \cos 17.2 + \beta_1 = 1 \qquad (11)$$

Solving simultaneously using Cramer's rule or determinant method, we have

$$\Delta_0 = \begin{vmatrix} 2\cos 40.1 & 1 \\ 2\cos 17.2 & 1 \end{vmatrix} = 2\cos 40.1 - 2\cos 17.2 = -0.3807$$

$$\Delta_{\beta_0} = \begin{vmatrix} 0 & 1 \\ 1 & 1 \end{vmatrix} = -1$$

$$\Delta_{\beta_1} = \begin{vmatrix} 2\cos 40.1 & 0 \\ 2\cos 17.2 & 1 \end{vmatrix} = 2\cos 40.1 = 1.5298$$

$$\beta_0 = \frac{\Delta_{\beta_0}}{\Delta_0} = \frac{-1}{-0.3807} = 2.6267 \quad ;$$

$$\beta_1 = \frac{\Delta_{\beta_1}}{\Delta_0} = \frac{1.5298}{-0.3807} = -4.0184$$

Substituting, we have $y(n) = \beta_0 x(n) + \beta_1 x(n-1) + \beta_0 x(n-2)$, the output becomes $y(n) = 2.6267x(n) - 4.0186x(n-1) + 2.6267x(n-2)$ (12)

OR y (n) =2.6267[x (n) +x (n-2)]−4.0184x (n−1) and the input becomes x (n) ={$\cos(0.3n) + \cos(0.7n)$}u(n)

Let, n=0:

$$y(n) = 2.6267x(n) - 4.0184x(n-1) + 2.6267x(n-2) \qquad (13)$$

$y(0) = 2.6267x(0) - 4.0184x(-1) + 2.6267x(-2)$; Initial conditions are x(-1) and x(-2).Note that x(0) is the first input value and not an initial condition.

$$n = 1: y(1) = 2.6267x(1) - 4.0184x(0) + 2.6267x(-1);$$

$$n = 2: y(2) = 2.6267x(2) - 4.0184x(1) + 2.6267x(0);$$

Step II: From the first principle, for high pass digital filter; In order to stop the high frequency component from appearing at the output of the filter, the magnitude function at ω_p =0.7 rad/time span should be equal to 1. Similarly, to pass the high frequency component within any attenuation, we need to ensure to ensure that the magnitude function at ω_s =0.3rad/s is equal to 0.

$$2\beta_0 \cos(0.7) + \beta_1 = 1 \qquad (14)$$

$$2\beta_0 \cos(0.3) + \beta_1 = 0 \qquad (15)$$

Or converting to degrees, we have

$$2\beta_0 \cos 40.1 + \beta_1 = 1 \qquad (16)$$

$$2\beta_0 \cos 17.2 + \beta_1 = 0 \qquad (17)$$

Solving simultaneously using Cramer's rule or determinant method, we have

$$\Delta_0 = \begin{vmatrix} 2cos40.1 & 1 \\ 2cos17.2 & 1 \end{vmatrix} = 2\cos 40.1 - 2\cos 17.2 = -0.3807$$

$$\Delta_{\beta_0} = \begin{vmatrix} 1 & 1 \\ 0 & 1 \end{vmatrix} = 1$$

$$\Delta_{\beta_1} = \begin{vmatrix} 2cos40.1 & 1 \\ 2cos17.2 & 0 \end{vmatrix} = -2cos17.2 = -1.9106$$

$$\beta_0 = \frac{\Delta_{\beta_0}}{\Delta_0} = \frac{1}{-0.3807} = -2.6267$$

$$\beta_1 = \frac{\Delta_{\beta_1}}{\Delta_0} = \frac{-1.9106}{-0.3807} = 5.0186$$

Substituting, we have $y(n) = \beta_0 x(n) + \beta_1 x(n-1) + \beta_0 x(n-2)$, the output becomes: $y(n) = -2.6267x(n) + 5.0186x(n-1) - 2.6267x(n-2)$ (18)

OR y (n) = −2.6267[x (n) +x (n-2)] +5.0186x (n-1) and the input becomes:

x (n) ={$\cos(0.3n) + \cos(0.7n)$}u(n)

let, n = 0;

$$y(n) = -2.6267x(n) + 5.0186x(n-1) - 2.6267x(n-2);$$

$$y(0) = -2.6267x(0) + 5.0186x(-1) - 2.6267x(-2)$$

; Initial conditions are x(-1) and x(-2).Note that x(0) is the first input value and not an initial condition.

$$n = 1: y(1) = -2.6267x(1) + 5.0186x(0) - 2.6267x(-1);$$

$$n = 2: y(2) = -2.6267x(2) + 5.0186x(1) - 2.6267x(0);$$

Initial conditions are $x(-1)$ and $x(-2)$.

The initial conditions are set to zero to satisfy linear and time-invariant properties.

IV. SYSTEM DESIGN ANALYSIS

The main goal of this design is to pass a desired frequency component and block another frequency component. In the design, the first 200 output samples were passed in both low pass and high pass filtering; the input sequence, x(n)=cos(0.3*n)+cos(0.7*n) was used to obtain the output sequence(n)=β_0x(n)+ β_1x(n-1)+ β_0x(n-2) where the filtering coefficient $\beta_0 = 2.6267$ and $\beta_1 = -4.0184$ for Low pass Digital FIR filter and $\beta_0 = -2.6267$ and $\beta_1 = 5.0186$ for High pass Digital FIR filter using different design techniques, employing Parks-McClellan Algorithm in the design.

4.1 Frequency Response of the Optimized Digital FIR Filter

The frequency responses for the optimized Low pass and High pass Digital FIR filter are designed using 45-order and 52-order respectively of optimized parks-McClellan Algorithm as shown in Appendix A and Appendix B. The cut-off frequencies of 0, 0.3, 0.7 and 1 were used assigning the desired amplitude of 1,1,0,0 and 0,0,1,1 for the Low pass and High Pass Digital FIR Filter respectively; generating the filter coefficient using three design tool box viz: filter design tool box(firgr), signal processing tool box(firpm) and filter visualization and tool box(fvtool) thereby allowing a Low frequency component of 0.3 and blocking a high frequency component of 0.7 for Low pass as shown in Figure 1 and allowing a High frequency component of 0.7 and blocking a low frequency component of 0.3 for High pass as shown in Figure 2

4.2. Impulse Response and convolution of Digital FIR Filters.

Appendix C and Appendix D show how to generate the impulse responses of second order Low pass and high pass Digital FIR FILTER were presented in Figure 3 and Figure 4 respectively, the two filter coefficient $\beta_{0,}$ and $\beta_{1,}$gotten from first principle as shown in equations above; coefficient $\beta_0 = 2.6267$ and $\beta_1 = -4.0184$ for Low pass Digital FIR filter and

$\beta_0 = -2.6267$ and $\beta_1 = 5.0186$ for High pass Digital FIR filter as well as the input sequence, x=cos (0.3*n) +cos (0.7*n) for the first 200 samples which were filtered to produce the output sequence. In Appendix E, combined impulse response was generated and the plot shown in Figure 5. Owing to finite length of Digital FIR Filter, in Appendix F, there was convolution of the both filters in frequency domain as plotted in Figure 6 below to show their filtering actions while their direct convolution is shown in Appendix G, Figure 7.

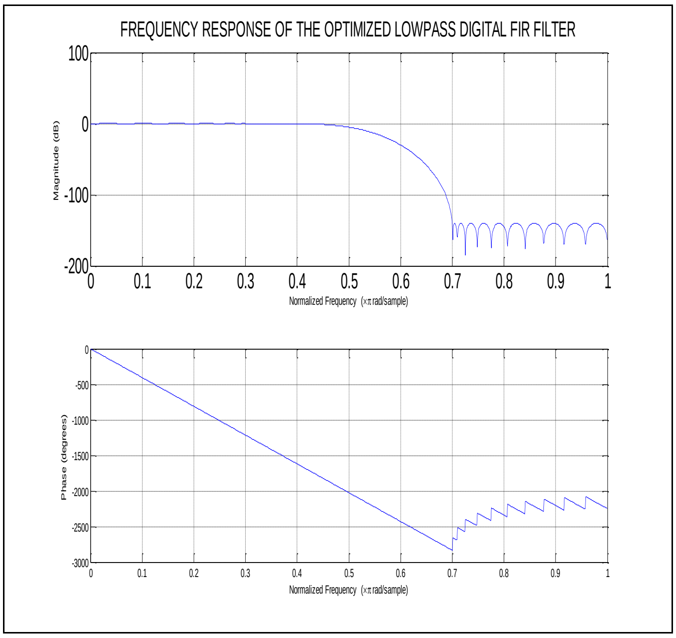


Fig. 1. Frequency Response of the Optimized Low pass Digital FIR Filter

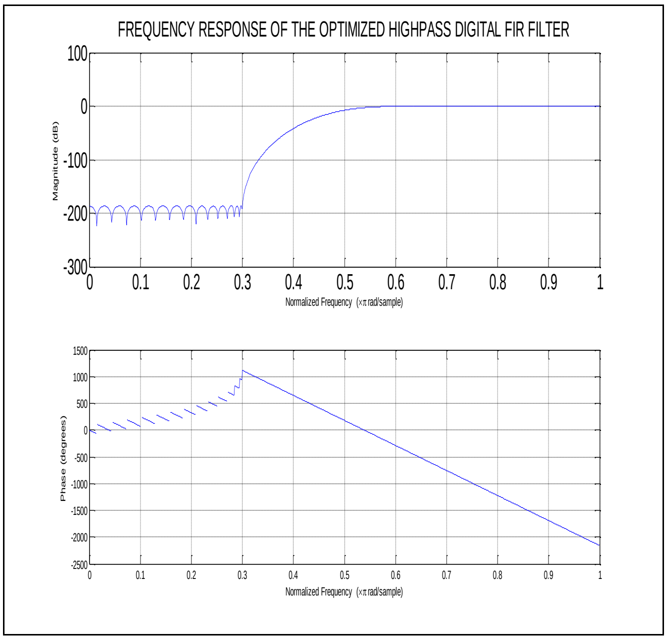


Fig. 2. Frequency Response of the Optimized High pass Digital FIR Filter

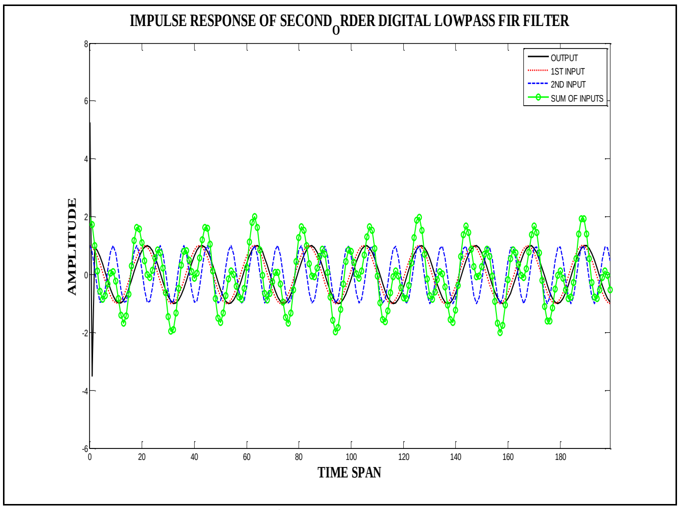


Fig. 3. Impulse Response of Second Order Low pass Digital FIR Filter

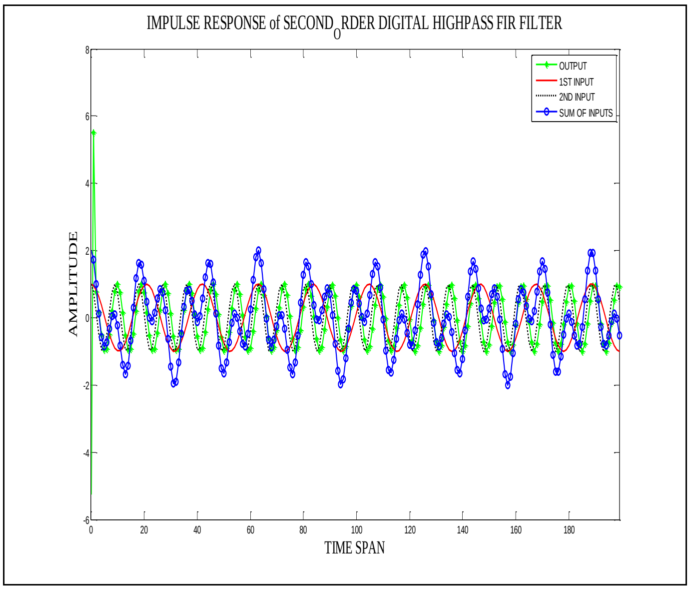


Fig. 4. Impulse Response of Second Order High Pass FIR Filter

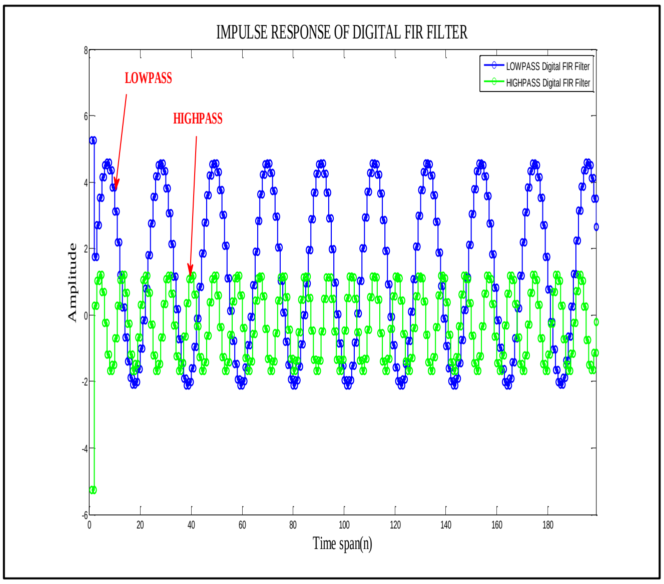


Fig. 5. Combined Impulse Responses of Low pass and High pass FIR Filters.

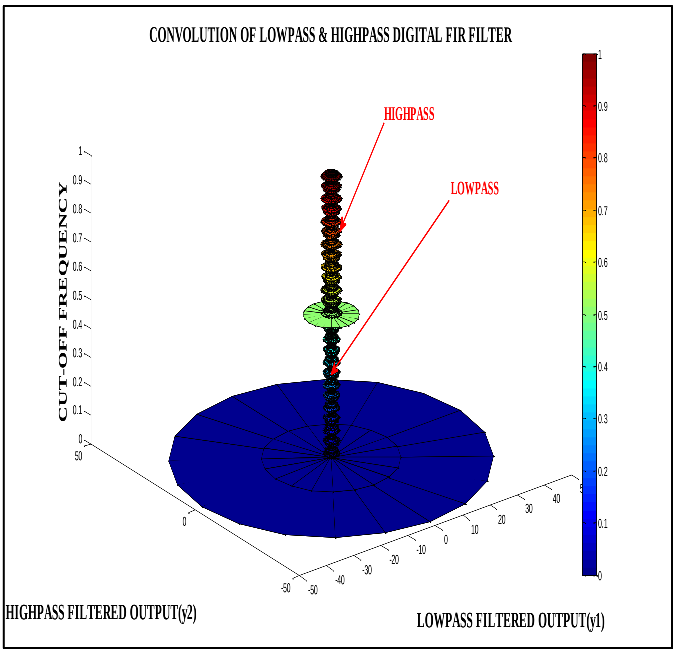


Fig. 6. Convolution of Digital FIR Filter

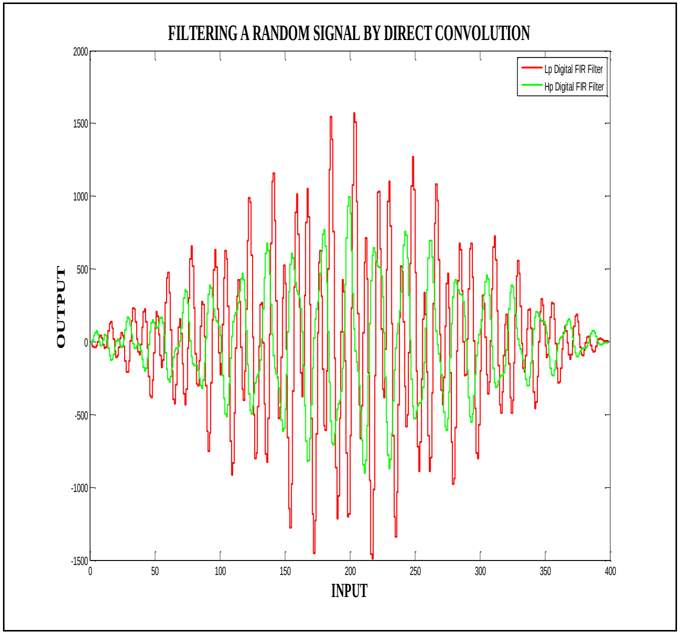
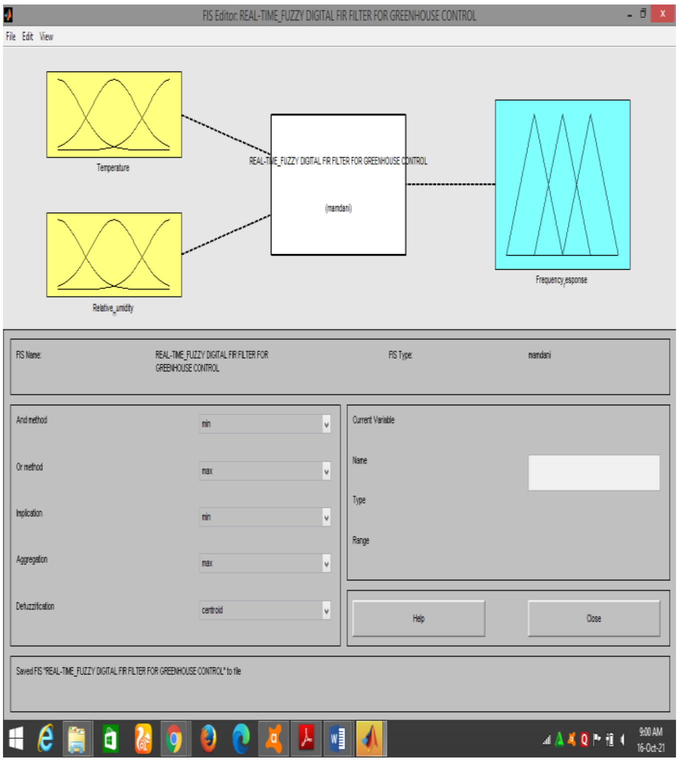


Fig. 7. Direct Convolution of Digital FIR Filter

4.3. Real-Time Fuzzy Control Modeling.

In the design of Low pass and high pass Digital FIR filter for the greenhouse control as shown in Figure 8−Figure 12, The fuzzy control system was built based on the inputs: Temperature and Humidity controlled by the Filter frequency response. We let the fuzzy variables Warm, Hot and Very Hot represent the Temperature Sensor and Low, Normal, High and very High represent the Humidity Sensors where the individual responses of the Digital FIR Filter for each combination of the inputs be Very Slow, Slow, Fast, Very Fast, and Extremely Fast. Gaussian membership function was used for both the linguistic input variable and the set of Output Control responses of the Digital FIR Digital filter. A defuzzifier collects the information provided by each of the rules and makes a decision from sentences defined by the inference engine about the output of models for a clear standard signal. The defuzzification priority procedure takes a weighted sum of the specified consequences of the rules according to the firing force of the rules. Fuzzy rules are a collection of linguistic statements that describe how the fuzzy controller should make a decision regarding the classification of an input or the control of an output. Fuzzy rules are always written as follows:

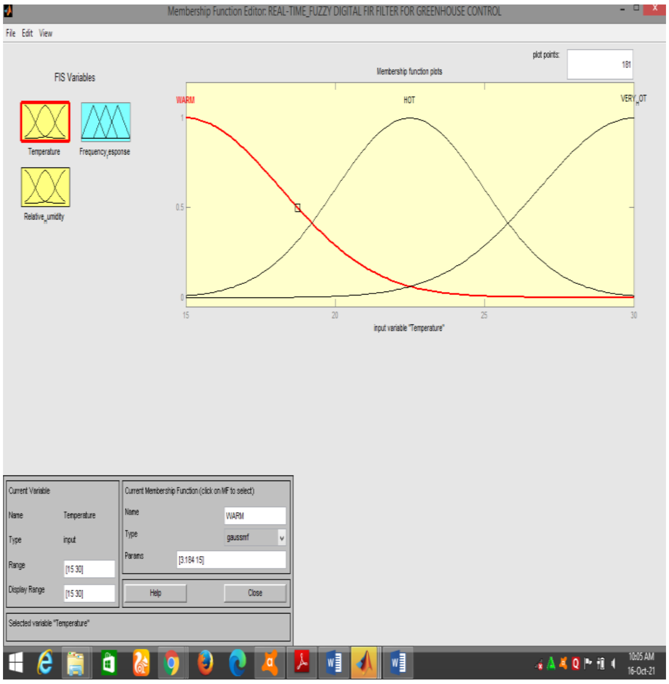
1. If (Temperature is Warm) and (Humidity is Normal) then (Frequency Response is Very Slow).
2. If (Temperature is Hot) and (Humidity is High) then (Frequency Response is Fast).
3. If (Temperature is Hot) and (Humidity is Low) then (Frequency Response is Slow).
4. If (Temperature is Hot) and (Humidity is Very High) then (Frequency Response is Very Fast).
5. If (Temperature is Very Hot) and (Humidity is High) then (Frequency Response is Extremely Fast).
6. If (Temperature is Very Hot) and (Humidity is Very High) then (Frequency Response is Extremely Fast).



FIS Name: REAL-TIME_FUZZY DIGITAL FIR FILTER FOR GREENHOUSE CONTROL
FIS Type: mandani
And method: min
Or method: max
Implication: min
Aggregation: max
Defuzzification: centroid

Saved FIS "REAL-TIME_FUZZY DIGITAL FIR FILTER FOR GREENHOUSE CONTROL" to file

Fig. 8. FIS Editor for Real-Time Fuzzy Digital FIR Filter

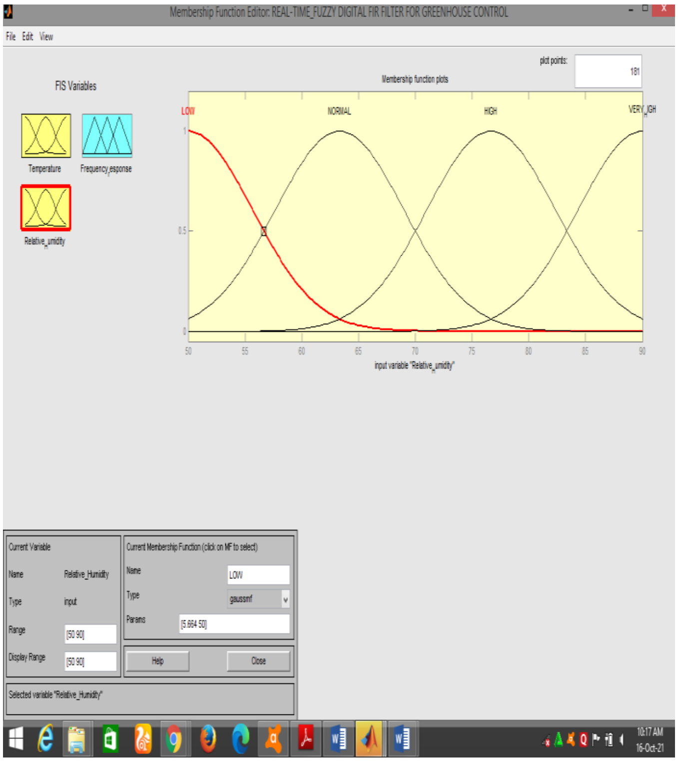


Current Variable
Name Temperature
Type input
Range [15 30]
Display Range [15 30]

Current Membership Function (click on MF to select)
Name WARM
Type gaussmf
Params [3.184 15]

Selected variable "Temperature"

Fig. 9. Membership function for linguistic input Variable “Temperature"

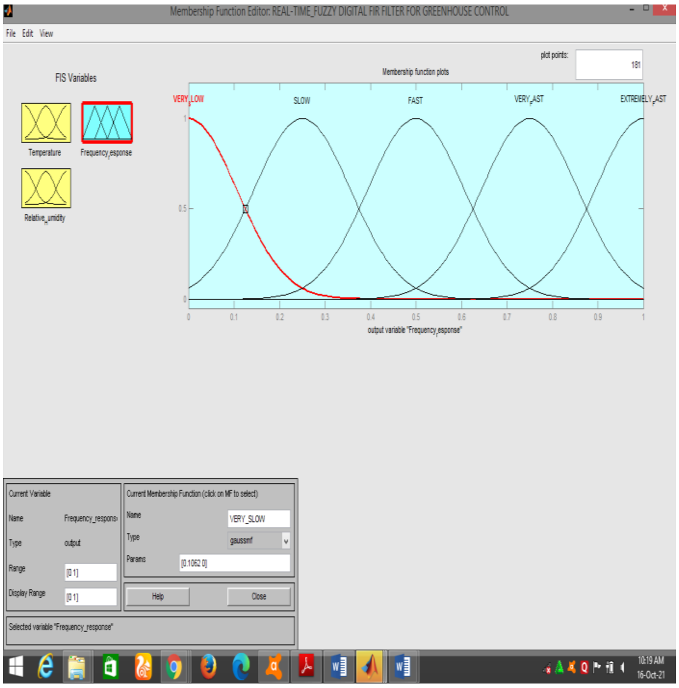


Current Variable
Name Relative_Humidity
Type input
Range [50 90]
Display Range [50 90]

Current Membership Function (click on MF to select)
Name LOW
Type gaussmf
Params [5.664 50]

Selected variable "Relative_Humidity"

Fig. 10. Membership function for linguistic input Variable “Humidity"

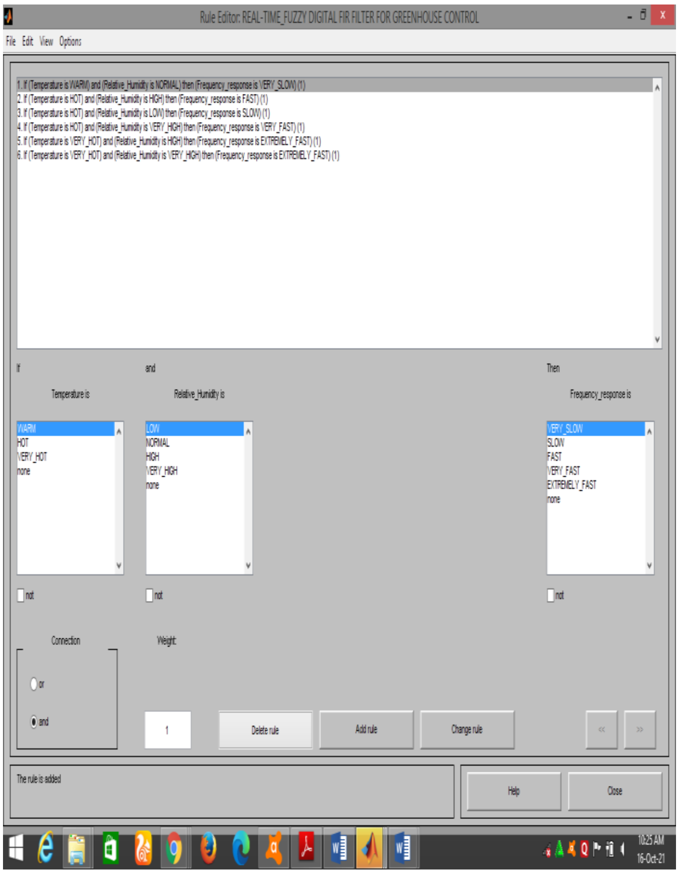


Current Variable
Name Frequency_respons
Type output
Range [0 1]
Display Range [0 1]

Current Membership Function (click on MF to select)
Name VERY_SLOW
Type gaussmf
Params [0.1062 0]

Selected variable "Frequency_response"

Fig. 11. Membership function for the control Output “Frequency Response”



1. If (Temperature is WARM) and (Relative_Humidity is NORMAL) then (Frequency_response is VERY_SLOW) (1)
2. If (Temperature is HOT) and (Relative_Humidity is HIGH) then (Frequency_response is FAST) (1)
3. If (Temperature is HOT) and (Relative_Humidity is LOW) then (Frequency_response is SLOW) (1)
4. If (Temperature is HOT) and (Relative_Humidity is VERY_HIGH) then (Frequency_response is VERY_FAST) (1)
5. If (Temperature is VERY_HOT) and (Relative_Humidity is HIGH) then (Frequency_response is EXTREMELY_FAST) (1)
6. If (Temperature is VERY_HOT) and (Relative_Humidity is VERY_HIGH) then (Frequency_response is EXTREMELY_FAST) (1)

Weight: 1

The rule is added

Fig. 12. Rule Editor for Real-Time Fuzzy Digital FIR Filter

As shown in table 1 below, in the analysis, temperature has universe of discourse 15 to 30°C with three degree of membership; humidity has universe of discourse 50 to 90% with four degree of membership; all the control output responses have 0 to 1 with five degree of membership for the next generation greenhouse control of fruits and vegetables nursery; done for every 30 seconds interval using temperature and Humidity Sensors. The Digital FIR Filters operate based on the performance of these sensors.

TABLE 1. Range and the number of membership function of fuzzy control parameters

Linguistic variables	**Universe of discourse**	**Number of Membership function**	**Fuzzy variable**	**Type of Membership Function**
Temperature	15−30℃	3	Warm Hot Very Hot	Gauss mf
Humidity	50−90%	4	Low Normal High High Very	Gauss mf
Frequency Response	0−1	5	Very Slow Slow Fast Very Fast Extremely Fast	Gauss mf

4.4. Fuzzy Real-Time Implementation of Digital FIR Filter for Greenhouse Control

The block diagram of the proposed Fuzzy Digital FIR Filter for Real-time Control Applications is shown in Figure 13. M-File MATLAB editor for Gaussian membership function of Digital FIR Filter was generated as shown in Figure 14, Appendix H. For the greenhouse control as shown in Table 2, Temperature and relative humidity ranges for optimal growth of various fruit and vegetable plants in a closer typical greenhouse where most fruit and vegetable plants require a temperature of 15-30 ° C and a relative humidity of 50-90 % for up to 180 samples.

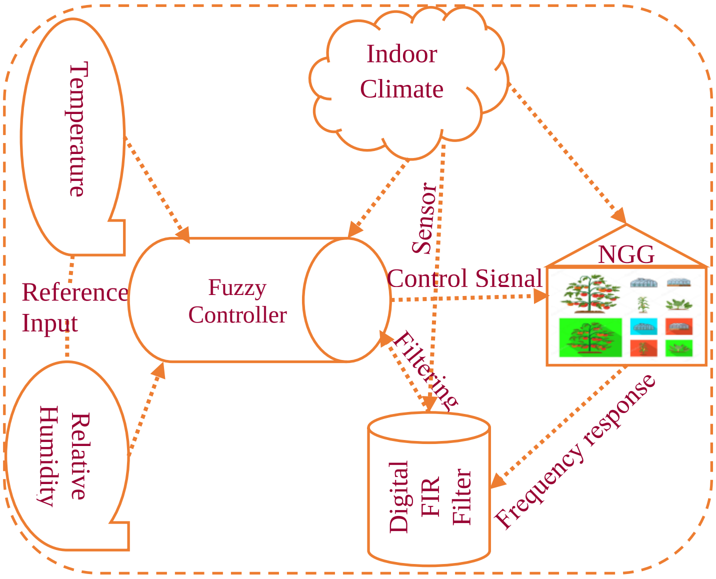


Fig. 13. The block diagram of Proposed Fuzzy Digital FIR Filter for Real-time Control Applications

The implementation of the greenhouse filtering and control of temperature and humidity is shown in Figure 15 and Figure 16, as illustrated in Appendix I and J. In Figure 17 and Figure 18, the rule viewer and surface viewer of the Real-time Fuzzy based Digital FIR Filter for Greenhouse control of next generation fruits and vegetables nurseries where average daily temperature of 22.5℃ and absolute relative humidity of 70% was achieved with sampled frequency of the greenhouse still maintained at 0.5π rads/s.

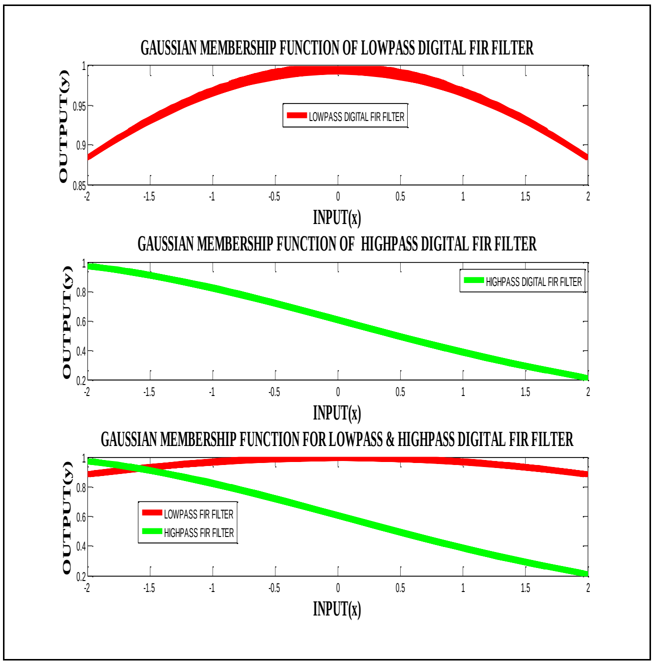


Fig. 14. Gaussian Membership Function of Digital FIR Filter

TABLE 2. Temperature and Humidity Greenhouse Control

Time Span(n)	**Temperature(℃)**	**Relative Humidity (%)**
30	15.0000	48.0959
60	18.0000	54.5246
90	21.0000	60.9533
120	24.0000	67.3820
150	27.0000	73.8107
180	30.0000	80.2394

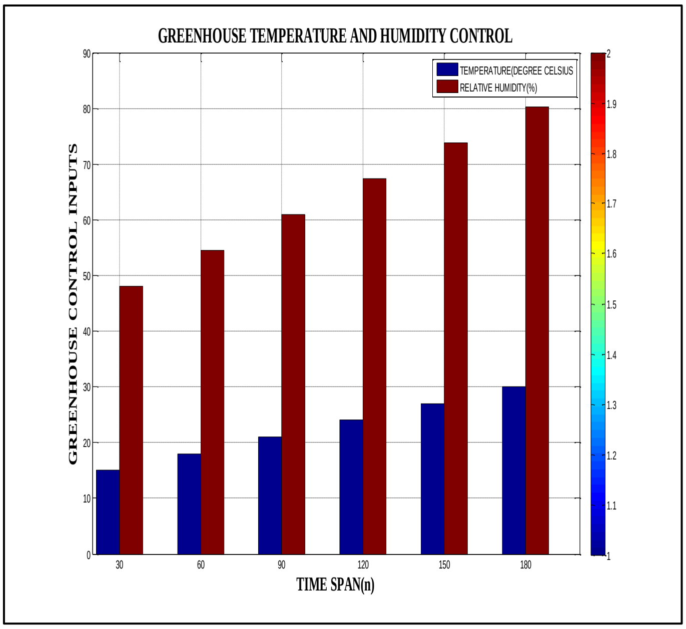


Fig.15. Next Generation Greenhouse Temperature and Humidity Control.

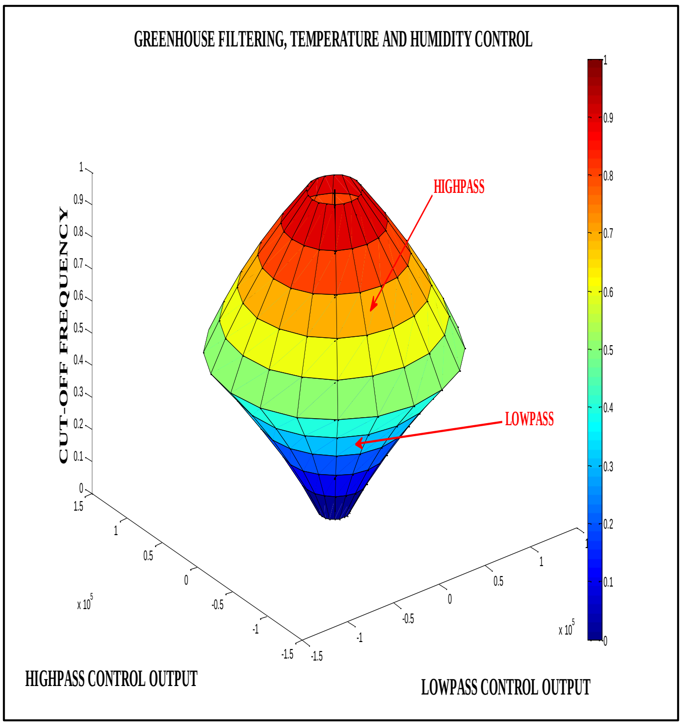
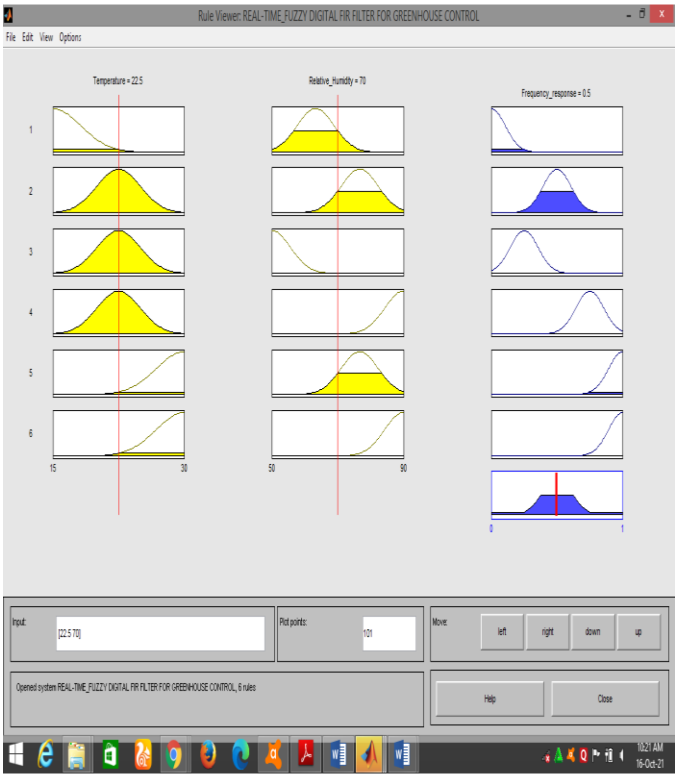
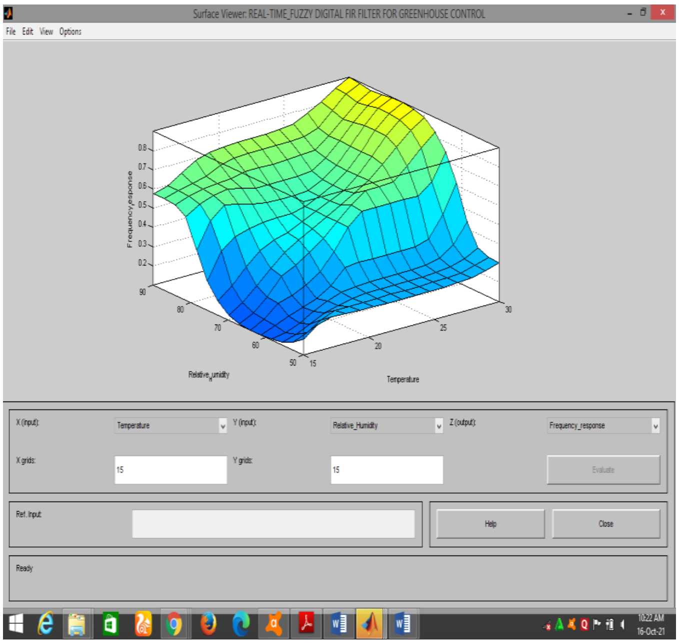


Fig. 16. Greenhouse FIR Digital Filtering & Control of Temperature and Humidity



Input: [22.5 70]
Plot points: 101
Move:
Opened system REAL-TIME_FUZZY DIGITAL FIR FILTER FOR GREENHOUSE CONTROL, 6 rules

Fig. 17. Rule Viewer of the Real-Time Fuzzy FIR Filter for Greenhouse Control



X (input): Temperature
Y (input): Relative_Humidity
Z (output): Frequency_response
X grids: 15
Y grids: 15
Ref. Input:
Ready

Fig. 18. Surface Viewer of the Real-Time Fuzzy FIR Filter for Greenhouse Control

V. CONCLUSION AND RECOMMENDATION FOR FUTURE STUDIES

5.1. Conclusion

Digital FIR filter is a non-recursive linear time invariant system with finite impulse response. The filter allowed a certain threshold of frequency to be passed while blocking the other. In the design, a Low pass Digital FIR Filter allowed a low frequency component of 0.3π rad/s and blocked a high frequency of 0.7π Rad/s component signals; a High pass Digital FIR Filter allowed a High frequency component of 0.7π Rad/s and blocked a low frequency of 0.3π rad/s component signals Using MATLAB editor by creating M-file in each case. Interestingly, it has no feedback but helps in fuzzy logic design in the design of a given control plant where an extremely fast frequency response is generated employing linguistic inputs to build a Gaussian membership functions. This makes the filter to perform a Fuzzy filtering. The stability and simplicity of the Digital FIR Filter help in easy optimization of the filter using MATLAB utilizing the filter design tool box (firgr), signal processing tool box (firpm) as well as the filter visualization and tool box (fvtool); employing the Parks-McClellan Algorithm, Window methods and Frequency sampling method. More so, the principle of Convolution technique was deployed to estimate the performance of the both filters in real-time greenhouse control of temperature of Humidity where daily average temperature of 22.5℃ and absolute relative humidity of 70% was achieved with cut-off frequency of the greenhouse for both the Lowpasss and High pass Digital FIR Filter still maintained at 0.5π rads/s.

5.2. Recommendation for Further Studies

As a future work, the use of fuzzy logic and Multi-rate systems in decision support system need to be more investigated and applied in various FIR filter based greenhouse control applications. Gaussian MF's are suitable for problems which require continuous differentiable curves and therefore smooth transitions, whereas the triangular do not possess these abilities. Therefore, the use of Gaussian membership functions instead of triangular membership functions currently used in the developed fuzzy system for students' evaluation need to be investigated and tested. The effect of using Digital FIR filter in improving the energy efficient in controlling greenhouses should be considered for further investigation. Other filters such as unscented bandpass filter, which is much more suitable for non-linear systems, need to be investigated and applied.

Appendix A

Frequency Response of The Optimized Low pass Digital FIR Filter

Create an M-file and save it with *LowPassFQ.m; then run the program*

```
%%===Design of The Frequency Response For optimized low
pass Digital FIR Filter Using Parks-McClellan Algorithm
 F1= [0, 0.3, 0.7, 1]; % Cut-off frequencies of the FIR DIGITAL
FILTER
 A1= [1, 1, 0, 0]; % Desired Amplitudes of the filter Design
 h1=firgr (45, F1, A1, [10, 1]); % filter Design @ 45-order
optimized Park-McClellan low pass digital FIR FILTER
 Subplot (2, 1, 1);
set(gca,'FontSize',18,'LineWidth',10,'FontWeight','bold','Font
Name', ‘Times New Roman')
 Freqz (h1, 1, 1024); % Computes and Plots the frequency
Response of Optimized LOWPASS DIGITAL FIR FILTER
 Title ('FREQUENCY RESPONSE OF THE OPTIMIZED
LOWPASS DIGITAL FIR FILTER');
 %end
```

OR

Using Signal Processing Tool Box, we have

```
h1=firpm (45, F1, A1, [10, 1]); % Compute the filter
Coefficient @ 45-order optimized Park-McClellan LOWPASS
DIGITAL FIR FILTER
 Freqz (h1, 1, 1024); % Computes and Plots the frequency
Response of Optimized LOWPASS DIGITAL FIR FILTER
 title ('FREQUENCY RESPONSE OF THE OPTIMIZED
LOWPASS DIGITAL FIR FILTER');
%end
```

OR

Using filter tool and visualization

```
b2=firls(45,[0,0.3,0.7,1],[1,1,0,0]);% compute filter
Coefficient @ 45-order of optimization design
H2=dfilt.dffir (b); % Filter Analysis
Fvtool (H2); % filter design visualization
%end
```

Appendix B

Frequency Response of The Optimized High pass Digital FIR Filter.

Create an M-file and save it with *HighPassFQ.m; then run the program*

```
%%%==== DESIGN OF THE FREQUENCY RESPONSE
FOR OPTIMIZED HIGHPASS DIGITAL FIR FILTER
USING Parks-McClellan Algorithm
%% HighPassFQ.m
 F2=[0,0.3,0.7,1];% Cut-off frequencies of the FIR DIGITAL
FIR FILTER
 Wp=0.7;% frequency passband
 Ws=0.3;% frequency stopband
 A2=[0,0,1,1];% Desired Amplitudes of the filter Design
 h2=firgr(52,F2,A2,[20,1]);% filter Design @ 52-order
optimized
Parks-McClellan HIGHPASS DIGITAL FIR FILTER
 subplot(2,1,1)
set(gca,'FontSize',18,'LineWidth',10,'FontWeight','bold','Font
Name','Times New Roman')
freqz(h2,1,1024);% Computes and plots the frequency
Response of
Optimized HIGHPASS DIGITAL FIR FILTER
 title('FREQUENCY RESPONSE OF THE OPTIMIZED
HIGHPASS DIGITAL FIR FILTER');
 %end
```

OR

Using Signal Processing Tool Box, we have

```
h2=firpm(52,F2,A2,[20,1]); ]);% Compute the filter coefficient
@ 52-order optimized Parks-McClellan HIGHPASS DIGITAL
FIR FILTER
freqz(h2,1,1024);% Computes and plots the frequency
Response of Optimized HIGHPASS DIGITAL FIR FILTER
 title('FREQUENCY RESPONSE OF THE OPTIMIZED
HIGHPASS DIGITAL FIR FILTER');
%end.
```

OR

Using Filter tools and Visualization

```
b=firls(52,[0,0.3,0.7,1],[0,0,1,1],[20,1]);% compute filter
coefficient @ 52-order of optimization design
Hd=dfilt.dffir(b);% filter Analysis
 fvtool(Hd);% filter design visualization
%end
```

Appendix C

MATLAB Code Generation for Digital Low pass FIR Filter

Create an M-file and save it with *LOWPASSIMPULSE.m; then run the program*

```
Clear all
 % DESIGN OF SECOND_ORDER LOWPASS DIGITAL
FIR FILTER
 n=0:199; % Time span
 b= [2.6267 -4.0184 2.6267]; % The filter coefficients of Low
pass Digital FIR FILTER
 zi= [0 0]; %show initial conditions
 p1=cos (0.3*n); % cosine coefficient of LOWPASS Digital
FIR FILTER
 p2=cos (0.7*n); % cosine coefficient of HIGHPASS Digital
FIR FILTER
 P=p1+p2; % Sum of two cosine coefficients
 y=filter (b, 1, P, zi); % Generate the filter output sequence of
Low pass Digital FIR FILTER
 plot(n,y,'k-',n,p1,'r:',n,p2,'b--',n,P,'og-','LineWidth',2);% plot
the input and output sequences
 axis ([0 199 -6 8]);
```

```
ylabel ('AMPLITUDE','FontSize',
16,'FontWeight','bold','FontName','Times New Roman');
xlabel ('TIMESPAN','FontSize',
16,'FontWeight','bold','FontName','Times New Roman');
 legend('OUTPUT','1ST INPUT','2ND INPUT','SUM OF
INPUTS','FontSize',16,'FontWeight','bold','FontName','Times
New Roman');
 title('IMPULSE RESPONSE OF SECOND_ORDER
DIGITAL LOWPASS FIR
FILTER','FontSize',18,'FontWeight','bold','FontName','Times
New Roman');
 %end.
```

Appendix D

MATLAB Code Generation for Impulse Response of Digital High pass FIR Filter

Create an M-file and save it with *HIGHPASSIMPULSE.m; then run the program*

```
clear all
 % Design of SECOND_ORDER Highpass Digital FIR
FILTER
 %HIGHPASSIMPULSE.m
 n=0:199;% Time span
 b=[-2.6267 5.0186 -2.6267];% The filter coefficients of
Highpass Digital FIR FILTER
 zi=[0 0];% show initial conditions
 c1=cos(0.3*n);% cosine coefficient of Lowpass Digital FIR
FILTER
 c2=cos(0.7*n);% cosine coefficient of Highpass Digital FIR
FILTER
 C=c1+c2;% Sum of two cosine coefficients
 y=filter(b,1,C,zi);% Generate the filter output of Highpass
Digital FIR FILTER
 plot(n,y,'g*-',n,c1,'r-',n,c2,'k:',n,C,'ob-','LineWidth',2);% plot
the input and output sequences
 axis([0 199 -6 8]);
 ylabel('AMPLITUDE','FontSize',16,'FontName','Times New
Roman');
 xlabel('TIME SPAN','FontSize',16,'FontName','Times New
Roman');
legend('OUTPUT','1ST INPUT','2ND INPUT','SUM OF
INPUTS','FontSize',16, 'FontName','Times New Roman');
title('IMPULSE RESPONSE of SECOND_ORDER DIGITAL
HIGHPASS FIR FILTER','FontSize',18,'FontName','Times
New Roman');
% end.
```

Appendix E

MATLAB CODE Generation for Both Digital Low pass & High pass FIR Filter.

Create an M-file and save it with *COMBIMPULSE.m; then run the program*

```
clear all; close all;
 % Design of second_order Lowpass and Highpass Digital FIR
Filter;
 n=0:199;% Time span
 b1=[2.6267 -4.0184 2.6267];% filter coefficient of Lowpass
digital FIR Filter
 b2=[-2.6267 5.0186 -2.6267];% filter coefficient of highpass
digital FIR Filter
 zi=[0 0];% show initial conditions
x=cos(0.3*n)+cos(0.7*n);% input coefficient of Digital FIR
Filter
 y1=cumsum(x);
 y2=filter(b1,1,y1,zi);
 y3=filter(b2,1,y1,zi);
 stairs(y2,'ob-','LineWidth',2);
 hold on
 stairs(y3,'og-','LineWidth',2);
 legend('LOWPASS Digital FIR Filter','HIGHPASS Digital
FIR Filter','FontSize',16,'FontName','Times New Roman');
 axis([0 199 -6 8]);
xlabel('Timespan(n)','FontSize',16,'FontName','Times New
Roman'); ylabel('Amplitude','FontSize',16,'FontName','Times
New Roman');
 title('IMPULSE RESPONSE OF DIGITAL FIR
FILTER','FontSize',18,'FontName','Times New Roman');
 %end
```

Appendix F

MATLAB Program for Convolution of Digital FIR Filter.

```
%======CONVOLUTION OF DIGITAL FIR FILTER
 %%CONVOLUTION.m
 close all;
 n=0:199;% time span
 x=cos(0.3*n)+cos(0.7*n);% input sequence Digital FIR
FILTER
 b1=[2.6267 -4.0184 2.6267];% filter coefficient of LOWPASS
Digital FIR FILTER
 b2=[-2.6267 5.0186 -2.6267];% filter coefficient of
HIGHPASS Digital FIR FILTER
 y1=filter(b1,1,x);% filtered output of LOWPASS Digital FIR
FILTER
 y2=filter(b2,1,x);% filtered output sequence of HIGHPASS
Digital FIR FILTER
 z1=conv(y1,y2);% convoluted output of LOWPASS &
HIGHPASS Digital FIR FILTER
 cylinder(z1);
 title('CONVOLUTION OF LOWPASS & HIGHPASS
DIGITAL FIR FILTER','FontSize',16,'Fontname','Times New
Roman','FontWeight','bold');
 grid off
 xlabel('LOWPASS FILTERED OUTPUT(y1)','FontSize',16,
'Fontname','Times New Roman','FontWeight','bold')
 ylabel('HIGHPASS FILTERED
OUTPUT(y2)','FontSize',16,'Fontname','Times New
Roman','FontWeight','bold')
 zlabel('CUT-OFF
FREQUENCY','FontSize',16,'Fontname','Times New
Roman','FontWeight','bold')
% end
```

Appendix G

MATLAB Program for Direct Convolution of Digital FIR Filter.

Create an M-file and save it with *DIRECTCONV.m; then run the program*

```
%%DIRECTCONV.m
n=0:199;% Time span;
 x=cos(0.3*n)+cos(0.7*n);% input sequence of the Digital FIR
Filter
 a1=2.6267;%first filter coefficient of Lowpass Digital FIR
Filter
 b1=-4.0184;%second filter coefficient of Lowpass Digital FIR
Filter
 a2=-2.6267;% first filter coefficient of Highpass Digital
FIRFilter
 b2=5.0186;% second filter coefficient of Highpass Digital FIR
filter
 r1=(b1-a1).*rand(10,1)+a1;%random value of Lowpass
Digital FIR Filter.
 r2=(b2-a2).*rand(20,1)+a2;% random value of Highpass
Digital FIR Filter
 u1=filter(r1,1,x);%filtered Lowpass Digital FIR filter
 u2=filter(r2,1,x);%filtered Highpass Digital FIR Filter
 w1=conv(u1,x);% convoluted Lowpass Digital FIR Filter
 w2=conv(u2,x);% convoluted Highpass Digital FIR Filter
 stairs(w1,'r','LineWidth',2);
 hold on
 stairs(w2,'g','LineWidth',2);
 legend('Lp Digital FIR Filter','Hp Digital FIR
Filter','FontSize',16,'Fontname','Times New
Roman','FontWeight','bold')
 grid on
xlabel('INPUT','FontSize',16,'Fontname','Times New
Roman','FontWeight','bold');
ylabel('OUTPUT','FontSize',16,'Fontname','Times New
Roman','FontWeight','bold');
 title('FILTERING A RANDOM SIGNAL BY
DIRECTCONVOLUTION','FontSize',18,'Fontname','Times
New Roman','FontWeight','bold');
 grid off
% end
```

Appendix H

MATLAB Code for Gaussian Membership Function of Low Pass & High Pass Digital FIR Filter.

Create an M-file and save it with GAUSSIAN.m *; then run the program*

```
clear all;
 close all;
 %%====Design of LOWPASS Digital FIR FILTER USING
GAUSSIAN MEMBERSHIP FUNCTION
 % GAUSSIAN.m
 n=0:199;% Time span
 x=cos(0.3*n)+cos(0.7*n);% input of Digital FIR FILTER
 y1=gaussmf(x,[-4.0184 0 2.6267 2.6267]);% output of
LOWPASS
Digital FIR FILTER
 subplot(3,1,1);
 plot(x,y1,'r','LineWidth',5)
 xlabel('INPUT(x)','FontSize',16,'FontName','Times New
Roman','FontWeight','bold');
ylabel('OUTPUT(y)','FontSize',16,'FontName','Times New
Roman','FontWeight','bold');
 legend('LOWPASS DIGITAL FIR FILTER','FontSize',16,
'FontName','Times New Roman','FontWeight','bold');
 title('GAUSSIAN MEMBERSHIP FUNCTION OF
LOWPASS DIGITAL FIR
FILTER','FontSize',16,'FontName','Times New
Roman','FontWeight','bold');
 %%====Design of HIGHPASS Digital FIR FILTER USING
GAUSSIAN MEMBERSHIP FUNCTION
 y2=gaussmf(x,[-2.6267 -2.6267 0 5.0186]);% output of
HIGHPASS Digital FIR FILTER
 subplot(3,1,2);
 plot(x,y2,'g','LineWidth',5);
xlabel('INPUT(x)','FontSize',16,'FontName','Times New
Roman' ,'FontWeight','bold');
 ylabel('OUTPUT(y)','FontSize',16,'FontName','Times New
Roman','FontWeight','bold');
 legend('HIGHPASS DIGITAL FIR FILTER','FontSize',16,
'FontName','Times New Roman','FontWeight','bold');
 title('GAUSSIAN MEMBERSHIP FUNCTION OF
HIGHPASS DIGITAL FIR
FILTER','FontSize',16,'FontName','Times New Roman'
,'FontWeight','bold');
 subplot(3,1,3);
 plot(x,y1,'r',x,y2,'g','LineWidth',5);
 legend('LOWPASS FIR FILTER','HIGHPASS FIR FILTER',
'FontSize'
,16,'FontName','Times New Roman','FontWeight','bold');
 xlabel('INPUT(x)','FontSize',16,'FontName','Times New
Roman','FontWeight','bold');
ylabel('OUTPUT(y)','FontSize',16,'FontName','Times New
Roman','FontWeight','bold');
 title('GAUSSIAN MEMBERSHIP FUNCTION FOR
LOWPASS & HIGHPASS
DIGITAL FIR FILTER','FontSize',16,'FontName','Times New
Roman','FontWeight','bold');
% end
```

Appendix I

MATLAB Program for Greenhouse Temperature and Humidity Control.

Create an M-file and save it with GREENHOUSECONTROL2.m*; then run the program*

```
clc;
clear all;
close all;
%% GREENHOUSE TEMPERATURE AND RELATIVE
HUMIDITY CONTROL
%GREENHOUSECONTROL2.m
n=30:30:199;% time span
T=[15 18 21 24 27 30];% Temperature control
RH=2.1429*T+15.9524;% Relative Humidity
Y=[15 48.0959;18 54.5246;21 60.9533;24 67.3820;27
73.8107;30 80.2494];
bar(n,Y,'hist');
legend('TEMPERATURE(DEGREE CELSIUS','RELATIVE
HUMIDITY(%)');
xlabel('TIMESPAN(n)','FontSize',16,'FontWeight','bold','Font
Name','Times New Roman');
```

```
ylabel('GREENHOUSE CONTROL INPUTS','FontSize',16,'FontWeight','bold','FontName','Times New Roman');
 title('GREENHOUSE TEMPERATURE AND HUMIDITYCONTROL','FontSize',18,'FontWeight','bold','FontName','Times New Roman');
grid on
 %end.
```

Appendix J

MATLAB Program for Implementation of Greenhouse Filtering, Temperature and Humidity Control

Create an M-file and save it with GREENHOUSEFILTERING.m*; then run the program*

```
%% IMPLEMENTATION OF GREENHOUSE FILTERING, TEMPERATURE AND HUMIDITY CONTROL
%GREENHOUSEFILTERING.m
n=0:199; % Time span
x=cos (0.3*n) +cos (0.7*n);
b1= [2.6267 -4.0184 2.6267]; % The filter coefficients of Low pass Digital FIR FILTER
b2= [-2.6267 5.0186 -2.6267]; % The filter coefficients of High pass Digital FIR FILTER
y1=gaussmf(x, [-4.0184 0 2.6267]); % output of LOWPASS Digital FIR FILTER
y2=gaussmf(x, [-2.6267 0 5.0186]); % output of HIGHPASS Digital FIR FILTER
z1=conv (y1, y2); % convoluted output of LOWPASS & HIGHPASS Digital FIR FILTER
T= [15 18 21 24 27 30]; % Temperature control
RH=[48.0959 54.5246 60.9533 67.3820 73.8107  80.2494];% Relative Humidity
TK=T+273; % Absolute Temperature
z2=conv (TK, RH); % convoluted Inputs of Temperature & Humidity control
Cylinder (z1);
hold on
Cylinder (z2);
xlabel('LOWPASSCUT-OFF','FontSize', 16,'FontWeight','bold','FontName','Times New Roman');
ylabel('HIGHPASS CUT-OFF','FontSize',16,'FontWeight','bold','FontName','Times New Roman');
zlabel('FUZZY CONTROL INPUTS','FontSize',16,'FontWeight','bold','FontName','Times New Roman');
title('GREENHOUSE FILTERING, TEMPERATURE AND HUMIDITY CONTROL','FontSize',16,'FontWeight','bold','FontName','Times New Roman');
grid on
grid off
%end
```

REFERENCES

[1] Simon van Mourik, Peter J.M.van Beveren, Irineo L.lopez-cruz, Eldert J,van Henten., “Improving climate monitoring in greenhouse cultivation via model Based filtering”, Biosystems engineering 181.www.elsevier.com/ locate/issn/15375110. 2019. pp.40-51

[2] Kelvin Ndubuisi Nnamani, “Fuzzy Modeling of Automatic Greenhouse Control”, *International Research Journal of Modernization in Engineering Technology and Science*, 3(3) (2021), pp. 87-92

[3] Noopur Srivastava, Vandana Vikas Thakar, “Design Band Pass FIR Digital Filter for Cut off Frequency Calculation Using Artificial Neural Network”, *International Research Journal of Engineering and Technology (IRJET)*, 4(6) (2017). pp.1367-1374.

[4] X. Z. Gao, S. J. Ovaska, Xianjia Wang, “A Fuzzy Predictive Filter for Sinusoidal Signals with Time-Varying Frequencies”, *World Academy of Science, Engineering and Technology International Journal of Electronics and Communication Engineering*, 1(10) (2007). pp.1583-1587.

[5] Xiangdong Huang, Senxue Jing, Zhaohua Wang, “Yan Xu, Closed-Form FIR Filter Design Based on Convolution Window Spectrum Interpolation”, *IEEE Transactions on Signal Processing*, 64(5). (2016). PP.1173-1186.

[6] S u peng, “Design and Analysis of FIR filters based on MATLAB”, Degree Project Thesis.2013
45p

[7] R.M. Aguilar, V. Muñoz and Y. Callero, “Control Application Using Fuzzy Logic: Design of a Fuzzy Temperature Controller”, www.intechopen.com, 2012, pp. 380-396.

[8] Amer Ali Ammar, Dr. Mohamed. K. Julboub, Ahmed, “Digital Filter Design (FIR) Using Frequency Sampling Method”, University Bulletin 15(3) (2013). pp.39-48

[9] Ozlem Aplay, Ebubekir Erdem, “the control of greenhouses based on fuzzy logic using wireless sensor networks”, *International Journal of Computational Systems*, 12(1), 2019, pp.190-203.

[10] Engr.Dr.Obinna.S Oguejiofor. “Advanced Digital Signal Processing Lecture Note-ECE 810 for Masters Students, 2017/2018”, Unpublished.