DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING,
FACULTY OF ECE,
Rajshahi University of Engineering & Technology, Bangladesh

EEE - 3200 – Electrical & Electronic Circuit Simulation Sessional

Sessional Report

Student Workbook

Submitted by

Ashraf Al- Khalique
Roll: 1801171
Session: 2018-2019
Dept. of Electrical & Electronic Engineering,
Rajshahi University of Engineering and Technology.

Table of Contents

Experiment No. 01

1.1 Experiment Name

Introduction to MATLAB simulation

1.2 Objectives

- To learn about MATLAB tools and how they work
- To become acquainted with the Simulink platform and Simulink library
- To learn how to implement a system from an equation in the Simulink platform.

1.3 Theory

1.3.1 Simulink

Simulink is an application that allows you to simulate signals and dynamic systems. Simulink explores two stages: model definition and model analysis. It includes an interactive graphical environment and a collection of customizable block libraries that allow you to design, simulate, implement, and test a wide range of time-varying systems such as communications, controls, signal processing, video processing, and image processing.

Simulink contains toolboxes for developing, simulating, and analyzing communication systems. In addition, source coding, channel coding, interleaving, analog and digital modulation, equalization, synchronization, and channel modeling are all possible with Simulink.

1.3.2 Simulink Library

The Simulink Library Browser is the library where you can locate all the Simulink blocks. Simulink software contains a large library of functions that are often used in system modeling. These are some examples:

- Commonly Used Blocks
- Continuous
- Discontinuous
- Discrete
- Logic and Bit Operation
- Lookup Tables
- Math Operation
- Model Verification
- Mode-Wide Utilities
- Port & Subsystem
- Signal Attributes
- Signal Routing
- Sinks
- Sources
- User defined Functions
- Additional Math & Discrete

1.3.3 Common Block Library

- Math Operation
- Continuous
- Port & Subsystem

- Signal Routing
- Sink
- Sources

1.4 Apparatus

- MATLAB Simulink

1.5 Implementation in Problem

S_1= 1 sin ωt ; f= 50hz

S_2= Triangular signal; f= 500hz, $V_{p\text{-}p}$ = 0 to 1

S_3= Triangular signal; f= 500hz, $V_{p\text{-}p}$ = -1 to 0

$$Out_1 = \begin{cases} 1, & S_1 \geq S_2 \\ 0, & else \end{cases}$$

$$Out_2 = \begin{cases} 1, & S_1 \leq S_3 \\ 0, & else \end{cases}$$

$$Output = \begin{cases} Out_1, & S_1\, is\ (+ve) \\ Out_2, & else \end{cases}$$

1.6 Simulink Block Diagram

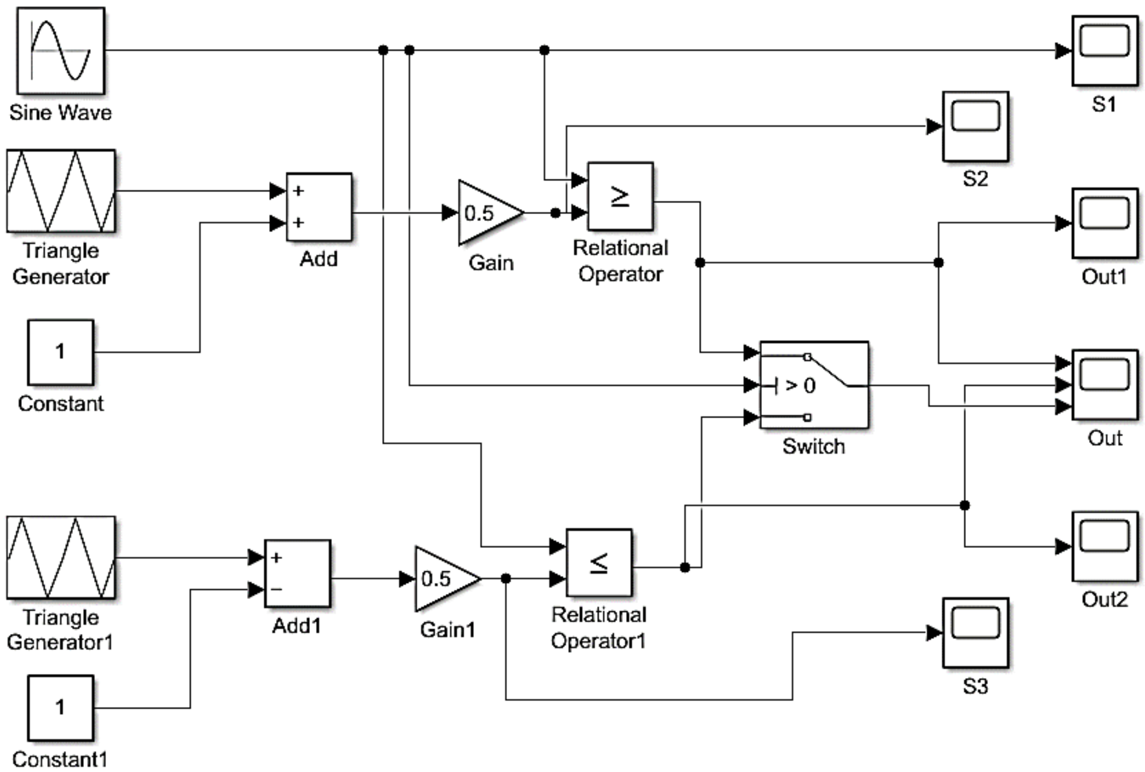


Fig 1.1: Circuit diagram for the implementation for the given problem

1.7 Graph

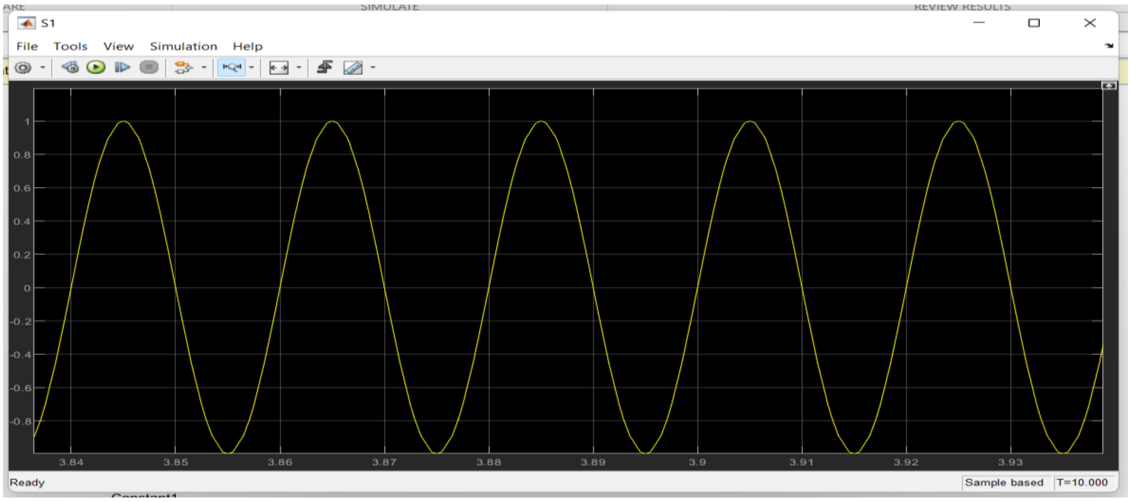


Fig 1.2: Sine wave signal (S1) (Amplitude = 1, f = 50 Hz)

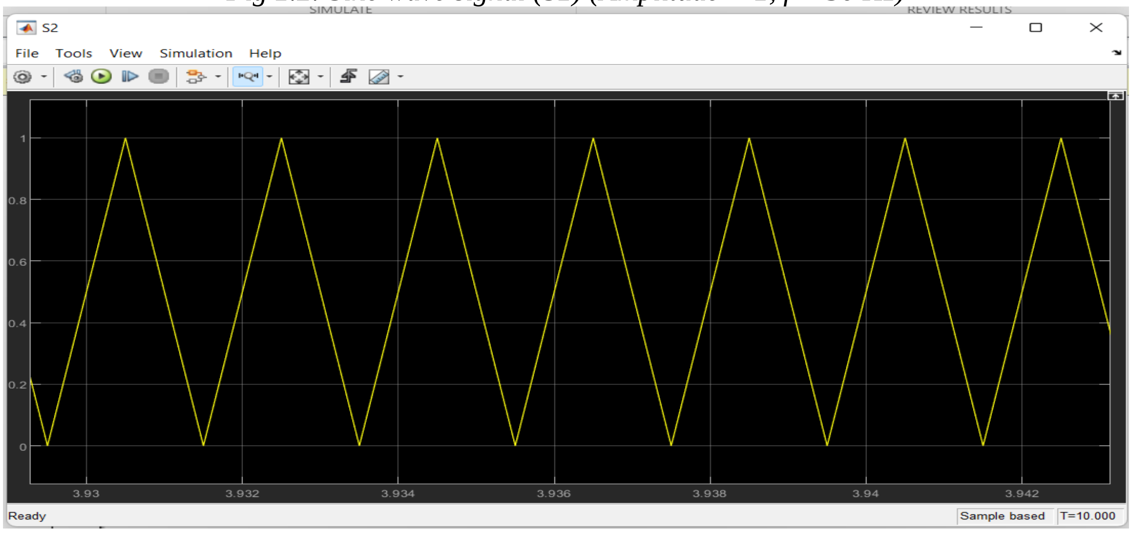


Fig 1.2: Triangle wave signal (S2); f= 500Hz, $V_{p\text{-}p}$ = 0 to 1

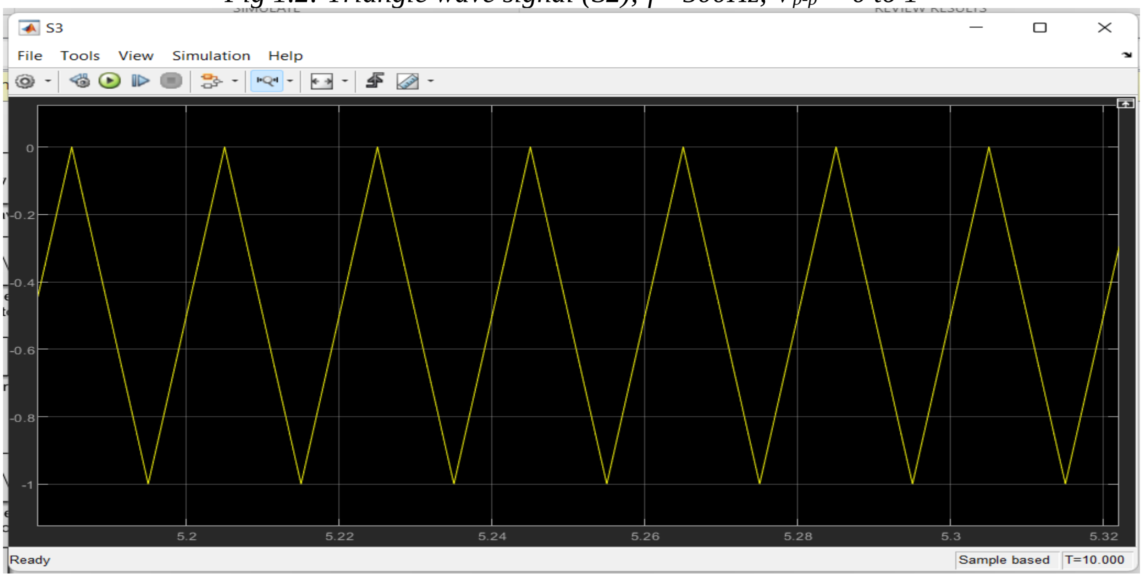


Fig 1.3: Triangle wave signal (S3); f= 500Hz, $V_{p\text{-}p}$ = -1 to 0

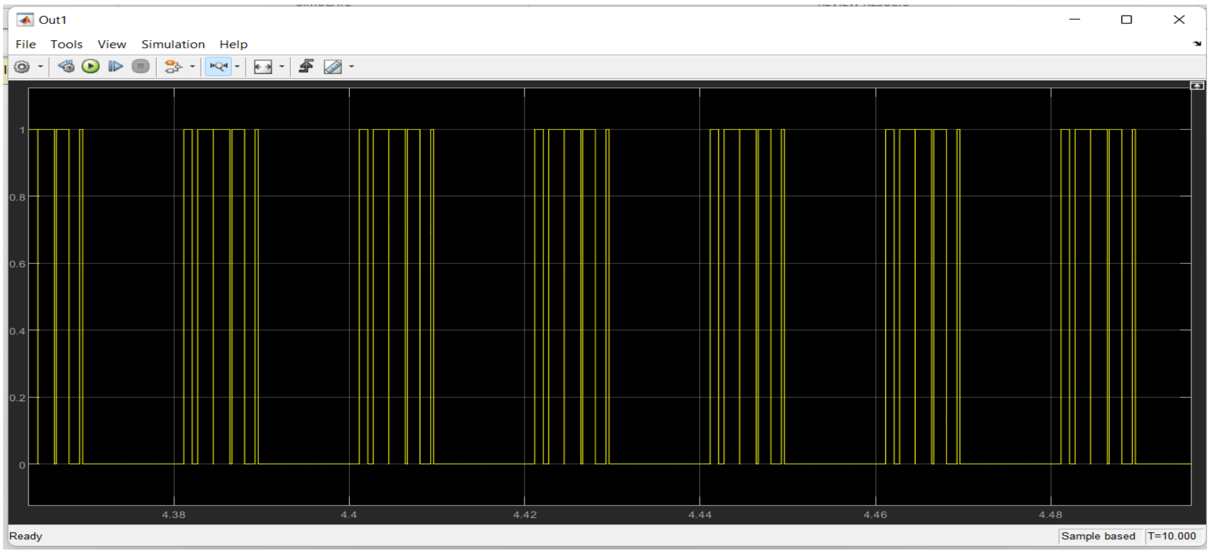


Fig 1.4: Output 1 wave, where 1 for, S1>=S2 else zero

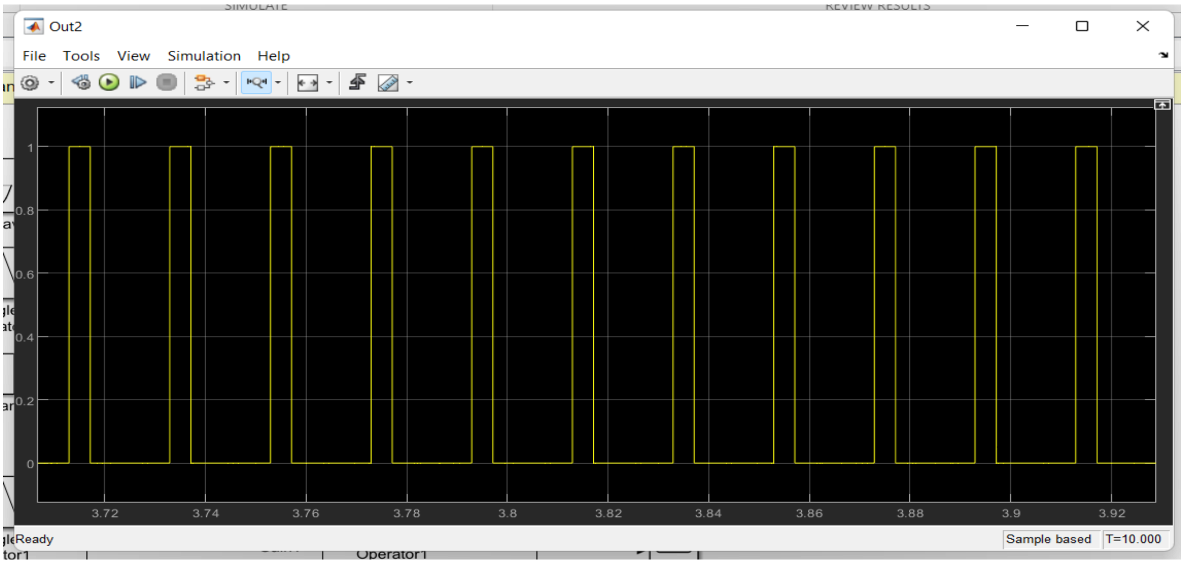


Fig 1.4: Output 2 wave, where 1 for, S1=<S2 else zero

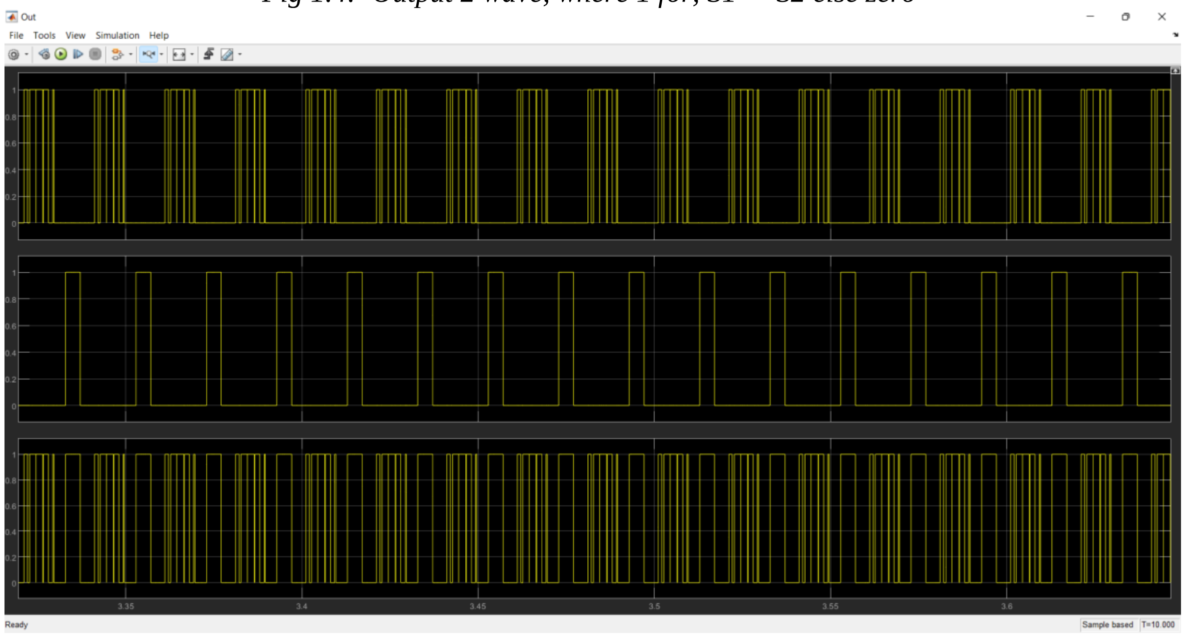


Fig 1.4: Output wave

1.8 Discussion & Conclusion

MATLAB Simulink is a powerful and flexible computational tool that is used extensively in academia and research worldwide.

This experiment's objective was to learn about Simulink tools, different types of library functions, and how to implement them. To understand its implementation method, a problem was solved. Here a sine wave and two triangular waves were used and compared using library functions. Thus, the final output was obtained and scoped in Simulink. As a result, the experiment is deemed a success.

Experiment No. 02

2.1 Experiment Name

Introduction to MATLAB Simulink Tools

2.2 Objectives

- To learn about MATLAB tools and how they work
- To become acquainted with various library functions
- To become acquainted with the Simulink platform and Simulink library
- To simulate various types of circuits using blocks from the library browser.

2.3 Theory

Simulink is a MATLAB-integrated simulation- and model-based design environment for dynamic and embedded systems. It is essentially a graphical block diagramming tool with a set of block libraries that can be customized.

For this experiment, following library functions are used in different circuits to observe their output.

2.3.1 MATLAB Function

The MATLAB Function blocks allow users to define custom functions in Simulink models using the MATLAB programming language. Simulink Coder and Embedded Coder can generate C/C++ code in MATLAB Function blocks. There are no continuous or discrete dynamic states in the custom functionality that you want to model.

2.3.2 Mean

The Mean block computes the input signal's mean value. The mean value is calculated over a one-cycle running average window of the specified fundamental frequency:

$$Mean(f(t)) = \frac{1}{T}\int_{(t-T)}^{t} f(t) \cdot dt$$

f(t): Input signal, T = 1/fundamental frequency

2.3.3 Absolute

The Abs block returns the input's absolute value. The absolute value of the most negative value is not representable by signed-integer data types. In this case, the Saturate on integer overflow check box governs the block's behavior. The Abs block can detect zero-crossings.

2.3.4 Sample and Hold

When a trigger event is received at the trigger port, the Sample and Hold block acquires the input at the signal port (marked by). The output is then held at the acquired input value until the next triggering event occurs.

2.3.5 Phase- Locked Loop (PLL)

A phase-locked loop is a feedback system that combines a voltage-controlled oscillator and a phase comparator to adjust the oscillator frequency or phase to track phase-modulated signal.

2.3.6 Half wave & Full wave rectifier

Half-wave rectifier is a circuit that only passes one half of the input sine wave. Full-wave rectifier is a circuit that converts both polarities of the input AC waveform to pulsating DC.

2.4 Apparatus

- Simulink

2.5 Simulink Block Diagram & Waveform

- **MATLAB Function**

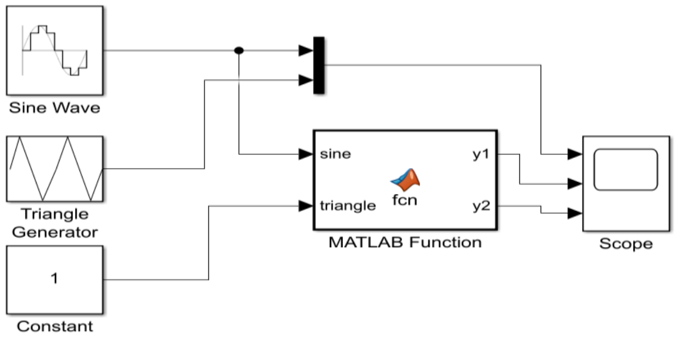


Fig 2.1: Block diagram with MATLAB function

- **Code**

```
function [y1, y2] = fcn(sine, triangle)
      %codegen
      S1=abs(sine)
      S2=(1+triangle)/2
      if S1>=0.5
            y1=1;
      else
            y1=0;
      end
      if S2>=S1
            y2=1;
      else
            y2=0;
      end
end
```

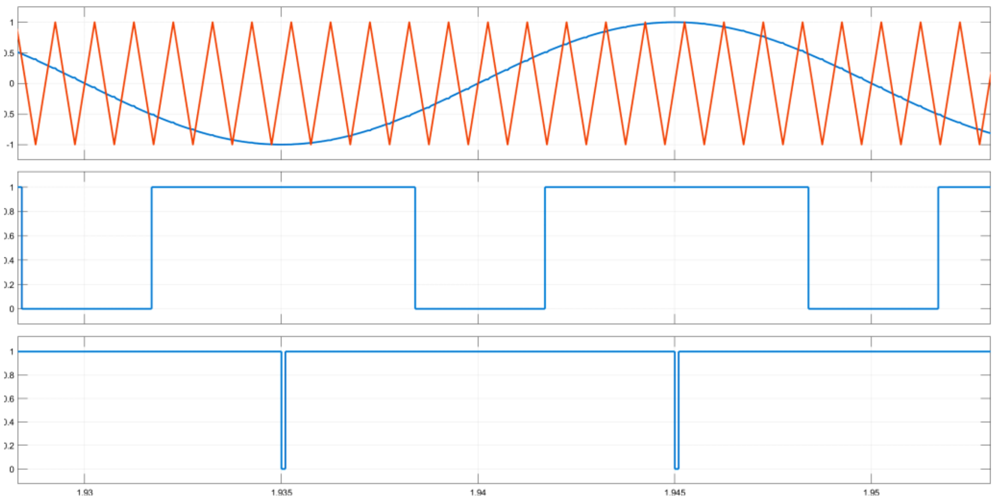


Fig 2.2: Input & Output waveform of sine and triangle wave using MATLAB function.

- **Mean**

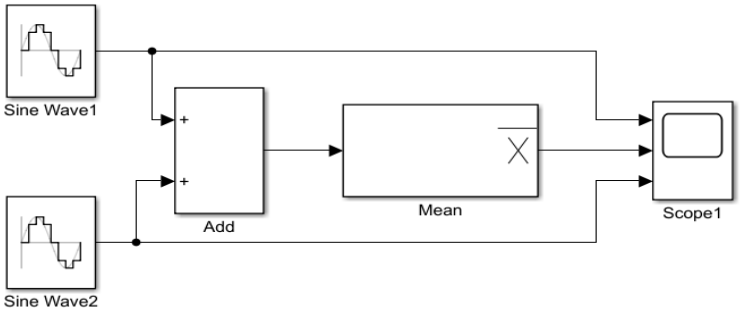


Fig 2.3: Block diagram with mean value

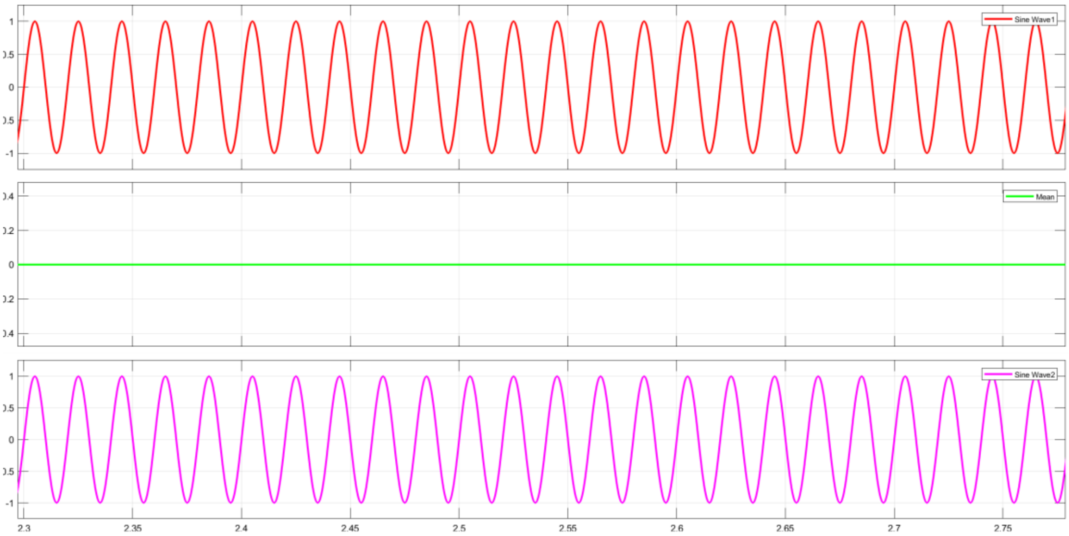


Fig 2.4: Input & Output waveform using Mean block function

- **Absolute Function**

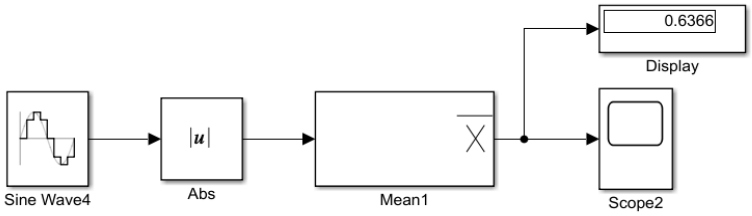


Fig 2.5: Block diagram with abs and mean value

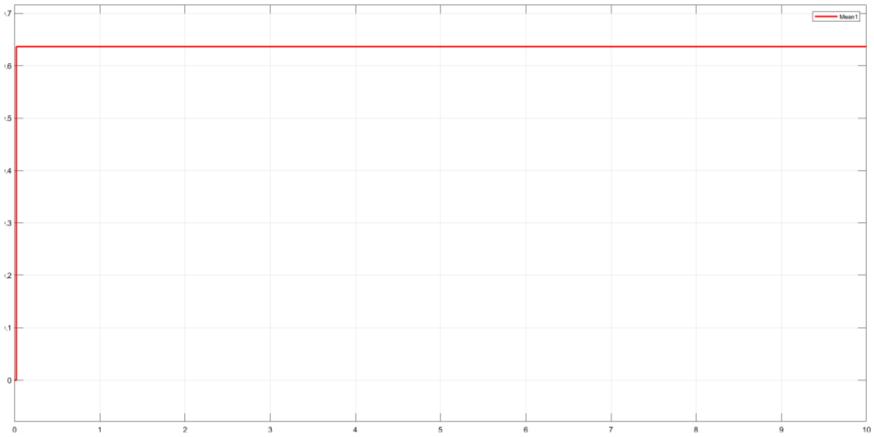


Fig 2.6: Output waveform using Abs block function

- **Sample and Hold**

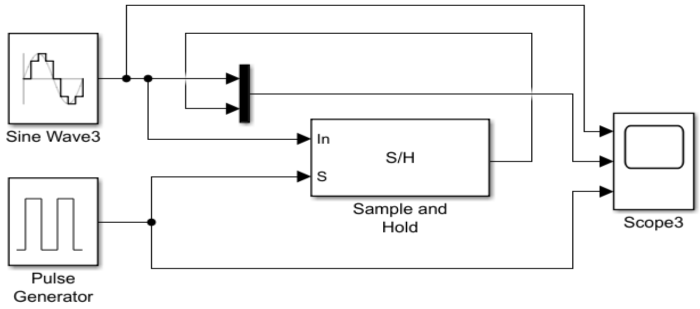


Fig 2.7: Block diagram of a sample and hold circuit

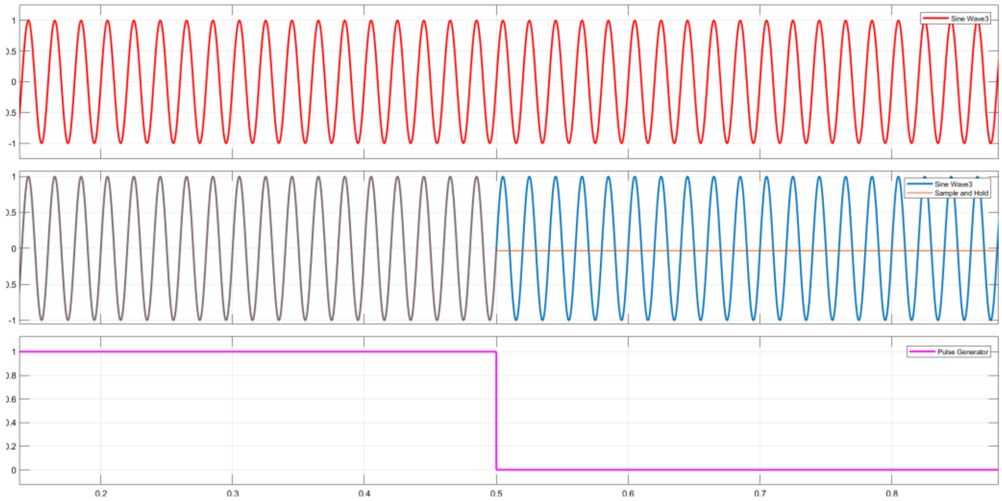


Fig 2.8: Input & Output waveform for Sample and Hold circuit

- **Phase- Locked Loop (PLL)**

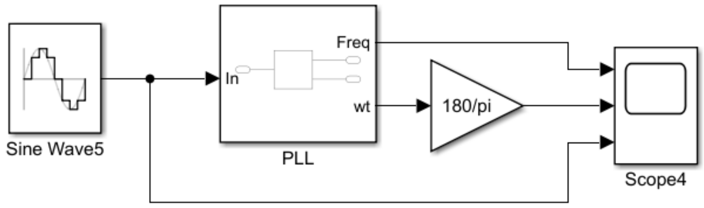


Fig 2.4: Block diagram of a PLL circuit

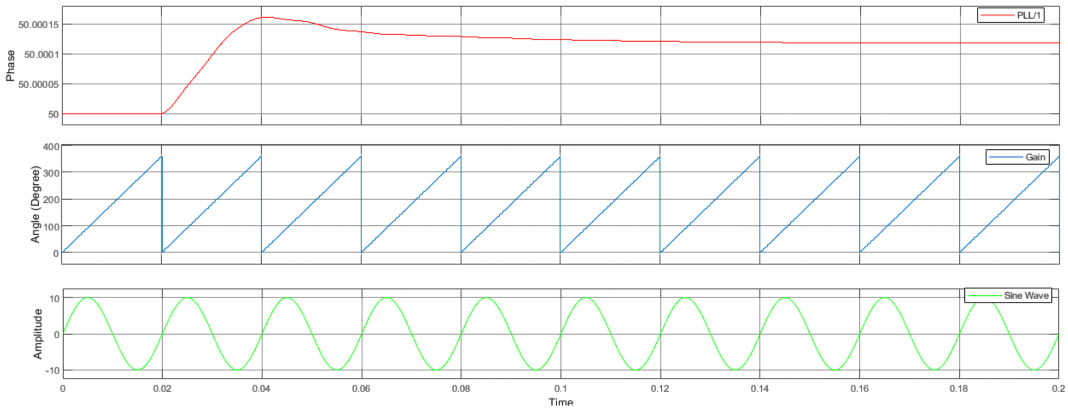


Fig 2.9: Input & Output waveform for Phase- Locked Loop circuit

- **PWM Generator**

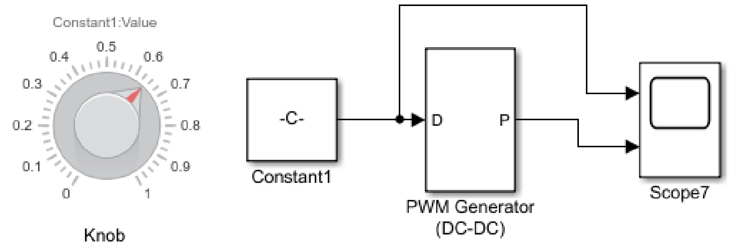


Fig 2.10: Block diagram of a PWM generator circuit

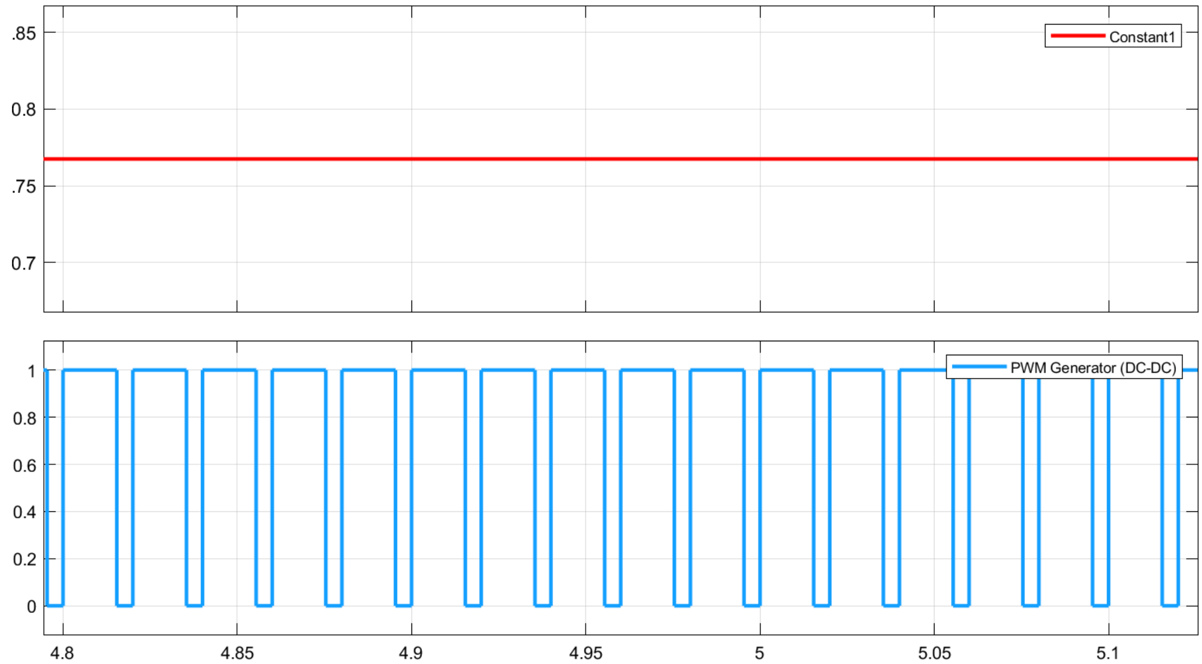


Fig 2.11: Constant & PWM waveform for PWM generator

- **Half-wave Rectifier**

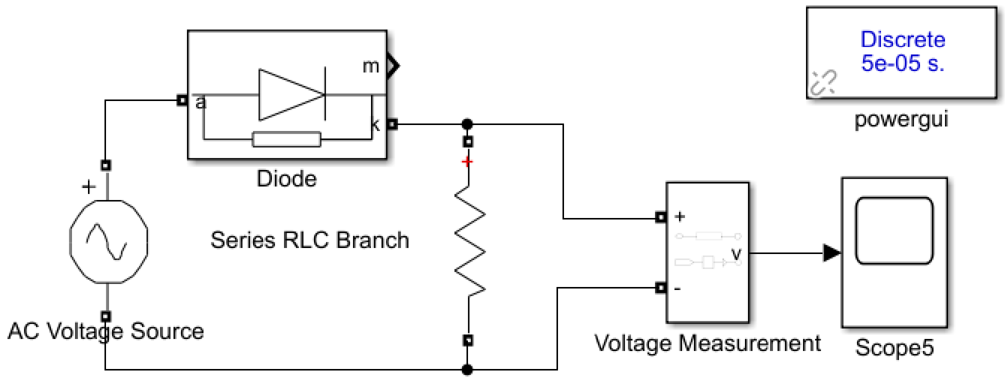


Fig 2.12: Block diagram of a half-wave rectifier circuit

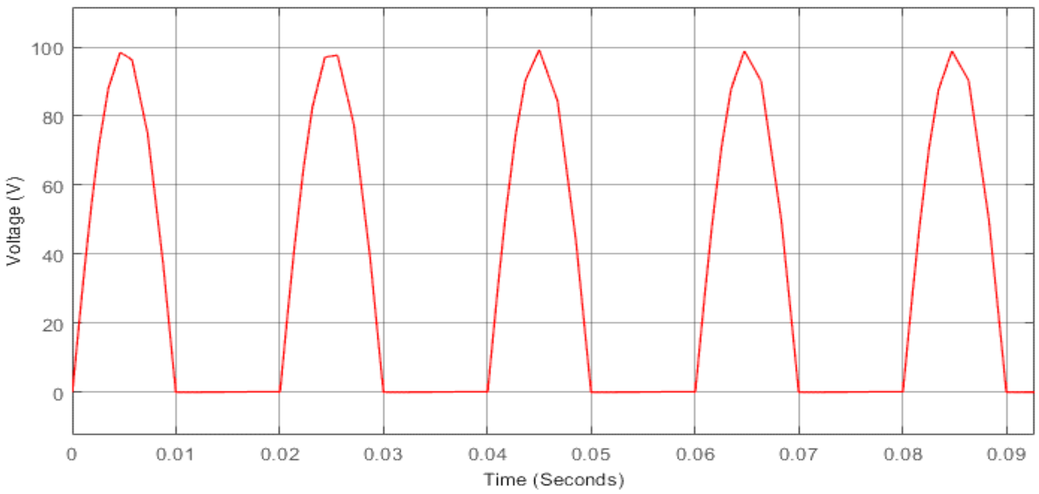


Fig 2.13: Output waveform using half-wave Rectifier circuit

- **Full-wave Rectifier**

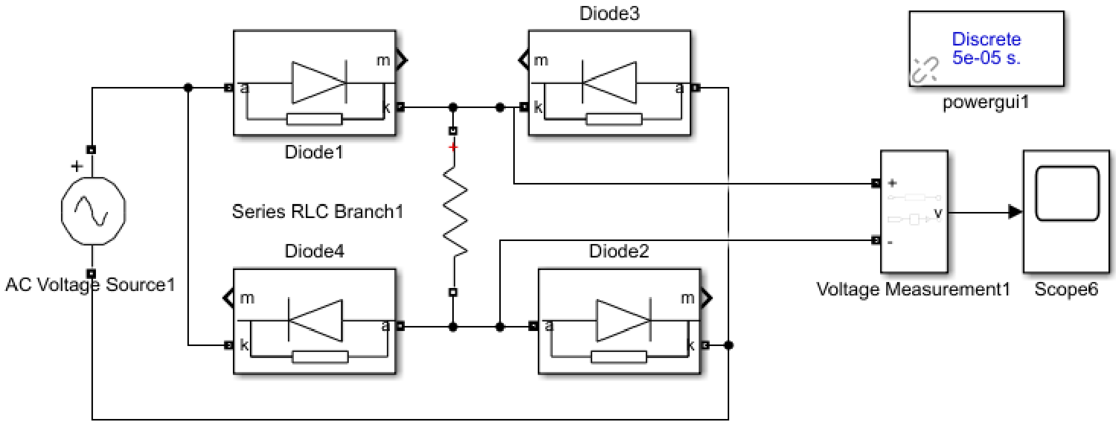


Fig 2.14: Block diagram of a full-wave rectifier circuit

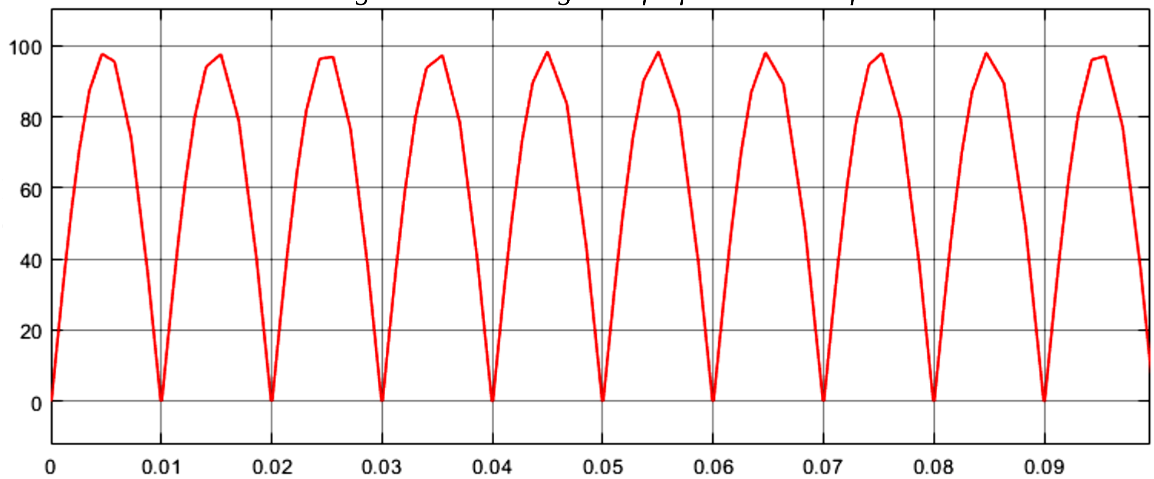


Fig 2.15: Output waveform using full-wave Rectifier circuit

2.6 Discussion & Conclusion

MATLAB Simulink is a powerful and flexible computational tool that is used extensively in academia and research worldwide.

Through this experiment, different types of blocks from the library function 'Simscape' from the library browser were introduced and implemented to PLL circuits, sample and hold circuits, PWM generator (DC-DC) and rectifier circuits (half-wave, full-wave). MATLAB function, mean block, and absolute block were also implemented in different circuit connections. In the end, outputs were viewed through a scope. As a result, the experiment was carried out correctly and the objectives were met.

Experiment No. 03

3.1 Experiment Name

Half-wave and Full-wave controlled rectifier circuits using Simulink

3.2 Objectives

- To learn about MATLAB tools and how they work
- To become acquainted with the Simulink platform and Simulink library
- To simulate various types of circuits using blocks from the library browser
- To design both half-wave and full-wave controlled rectifier circuits in Simulink

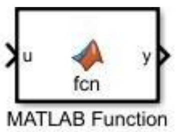
3.3 Theory

Simulink is a MATLAB-integrated simulation- and model-based design environment for dynamic and embedded systems.

For this experiment, half-wave and full-wave controlled rectifier circuits were designed following library functions and their output were observed,

3.3.1 MATLAB Function

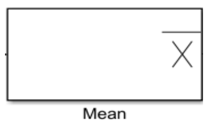
The MATLAB Function blocks allow users to define custom functions in Simulink models using the MATLAB programming language. There are no continuous or discrete dynamic states in the custom functionality that you want to model.



3.3.2 Mean

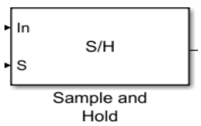
The Mean block computes the input signal's mean value. The mean value is calculated over a one-cycle running average window of the specified fundamental frequency:

$$Mean(f(t)) = \frac{1}{T}\int_{(t-T)}^{t} f(t)\cdot dt$$



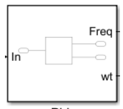
3.3.3 Sample and Hold

When a trigger event is received at the trigger port, the Sample and Hold block acquires the input at the signal port (marked by). The output is then held at the acquired input value until the next triggering event occurs.



3.3.4 Phase- Locked Loop (PLL)

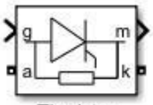
A phase-locked loop is a feedback system that combines a voltage-controlled oscillator and a phase comparator to adjust the oscillator frequency or phase to track phase-modulated signal.



PLL

3.3.5 Thyristor

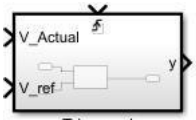
A thyristor is connected in parallel with a series RC snubber circuit. The thyristor model contains internal resistance and inductance when it is turned on. The internal inductance should be set to zero for most applications.



Thyristor

3.3.6 Triggered Subsystem

A triggered subsystem is a conditionally executed atomic subsystem that runs whenever the control signal (trigger signal) changes from a negative to a positive or zero value, or from zero to a positive value.



Triggered Subsystem

3.4 Apparatus

- Simulink

3.5 Simulink problem

The input voltage is 100sin(ωt), where the line frequency is 50 Hz. The circuit consists of a resistive load of 10Ω. Build a closed-loop control algorithm so that the output dc voltage remains constant at 40V. Show the input voltage, firing pulse, output voltage, and dc value waveforms in the scope. Simulate this in case of

- **Half-wave rectifier circuit**
- **Full-wave rectifier circuit**

3.6 Simulink Block Diagram & Waveform

- **Half-wave Rectifier**

1801171

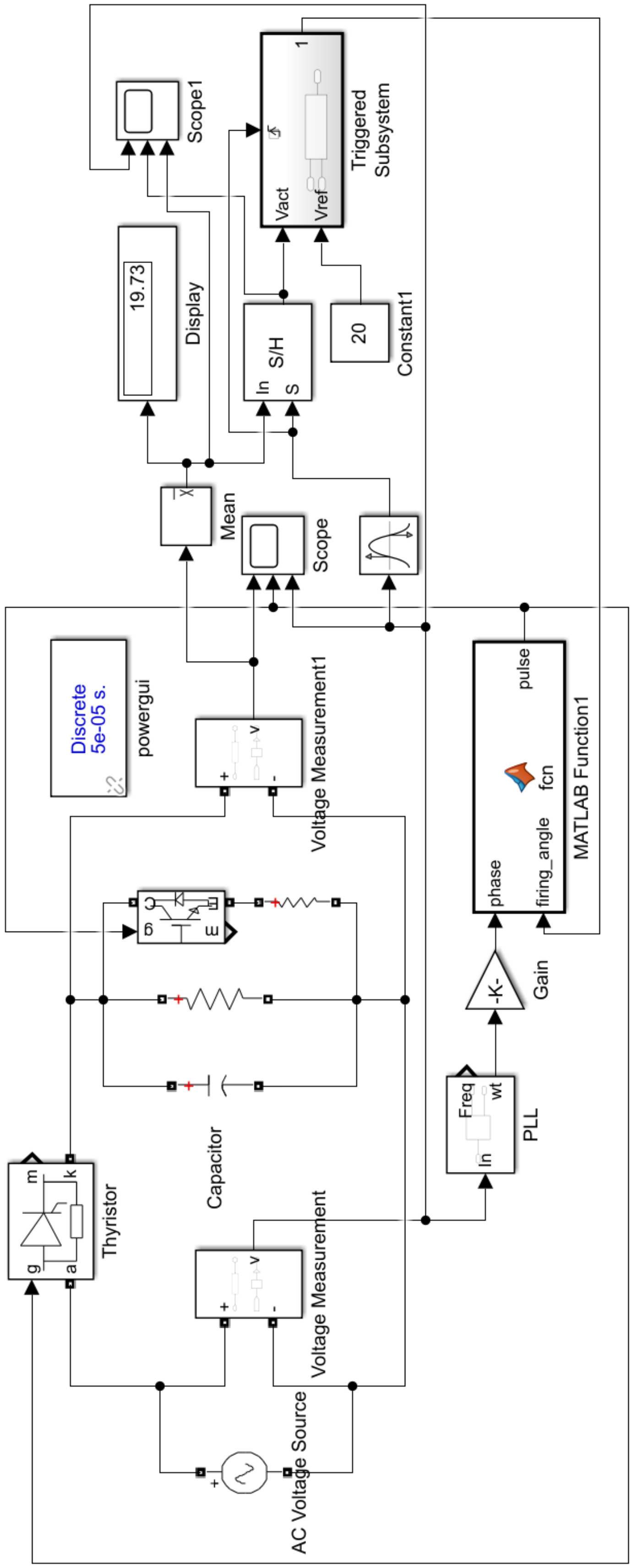


Fig. 3.1: Block diagram for controlled half- wave rectifier circuit

- **Code for triggered system**

```
function x = fcn(Vact,Vref)
%codegen
persistent firing_angle
if isempty(firing_angle),firing_angle=10,end;
step = 0.5;
if Vact<Vref
      firing_angle = firing_angle-step;
       if firing_angle<0
          firing_angle = 0;
       end
elseif Vact>Vref
         firing_angle = firing_angle+step;
          if firing_angle>180
             firing_angle = 180;
          else
              firing_angle = firing_angle;
          end
end
x = firing_angle;
end
```

- **Code for MATLAB function**

```
function pulse = fcn(phase,firing_angle)
%codegen
pulse_width = 10;
if
(phase>=firing_angle)&&(phase<=min((firing_angle+pulse_width),180))
      pulse = 1;
else
      pulse = 0;
end
end
```

- **Waveform**

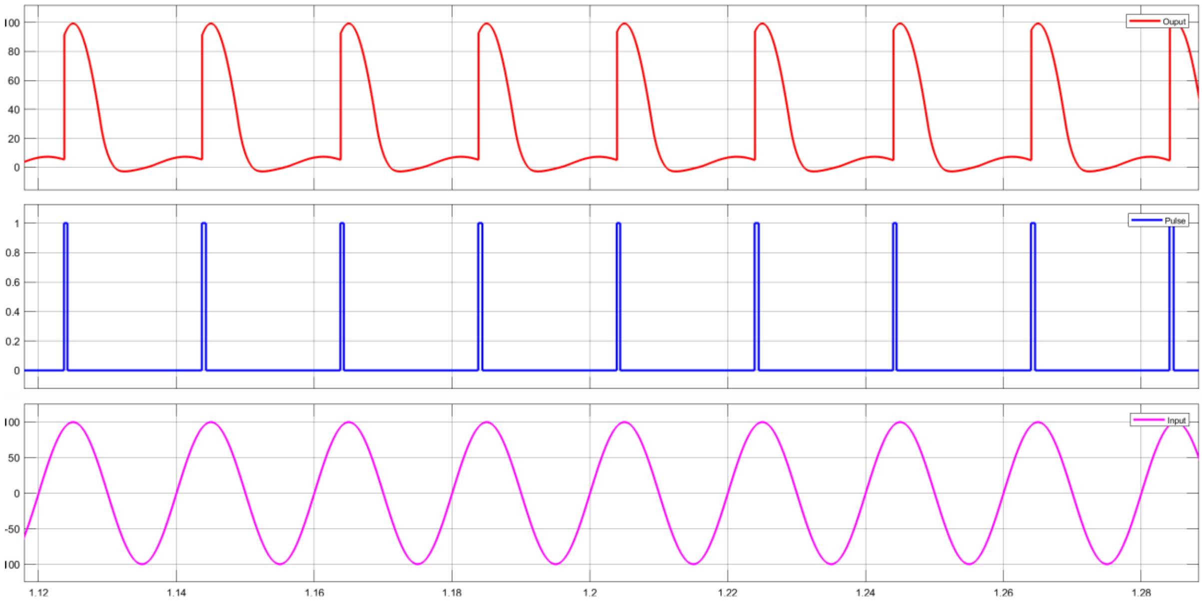


Fig. 3.2: Output, pulse, and input waveform for controlled half- wave rectifier circuit

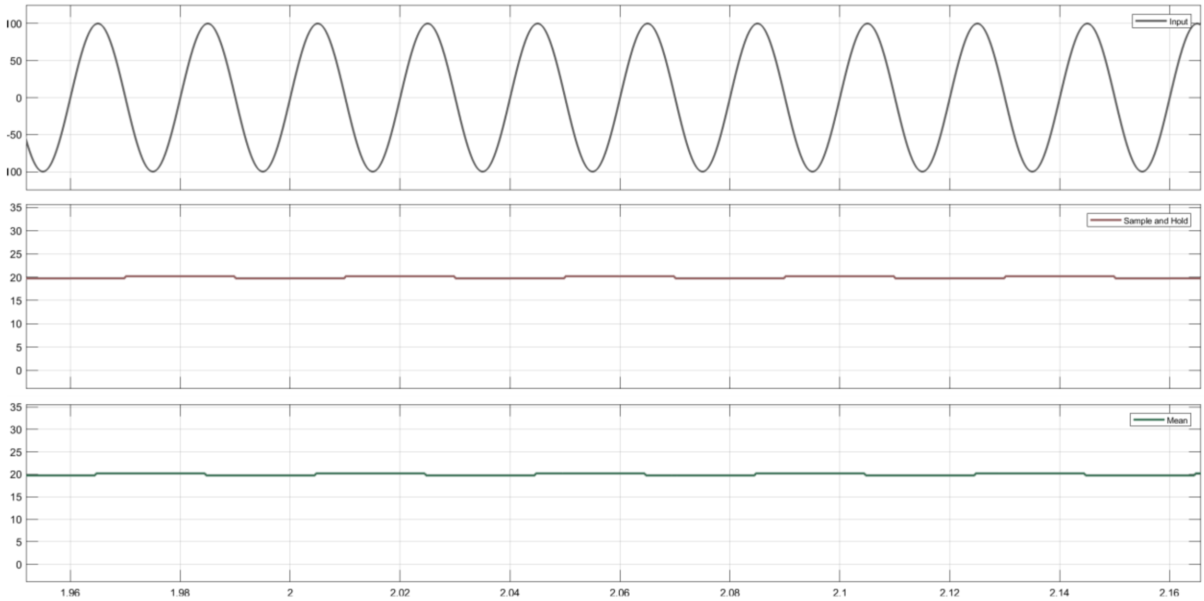


Fig. 3.3: Input, sample & hold, and mean waveform for controlled half- wave rectifier circuit

- **Full-wave Rectifier**

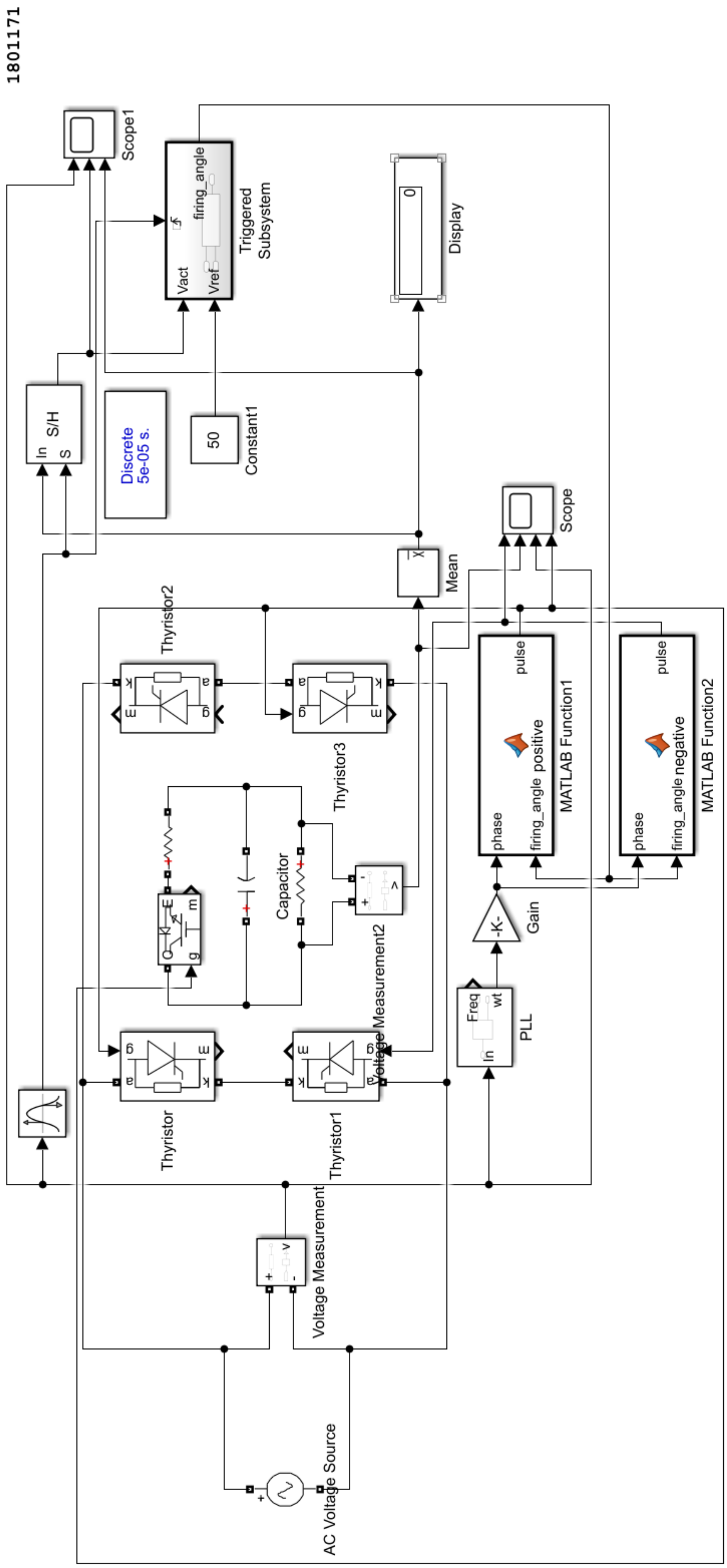


Fig. 3.4: Block diagram for controlled full- wave rectifier circuit

- **Code for triggered system**

```
function x = fcn(Vact,Vref)
%codegen
persistent firing_angle
if isempty(firing_angle),firing_angle=10,end;
step = 0.5;
if Vact<Vref
      firing_angle = firing_angle-step;
       if firing_angle<0
          firing_angle = 0;
       end
elseif Vact>Vref
         firing_angle = firing_angle+step;
          if firing_angle>180
             firing_angle = 180;
          else
              firing_angle = firing_angle;
          end
end
x = firing_angle;
end
```

- **Code for MATLAB function (positive half cycle)**

```
function pulse = positive(phase,firing_angle)
%codegen
pulse_width = 10;
if
(phase>=firing_angle)&&(phase<=min((firing_angle+pulse_width),180))
      pulse = 1;
else
      pulse = 0;
end
end
```

- **Code for MATLAB function (negative half cycle)**

```
function pulse = negative(phase,firing_angle)
%codegen
pulse_width = 10;
if
(phase>=(firing_angle+180))&&(phase<=min((firing_angle+pulse_width+1
80),360))
      pulse = 1;
else
      pulse = 0;
end
end
```

- **Waveform**

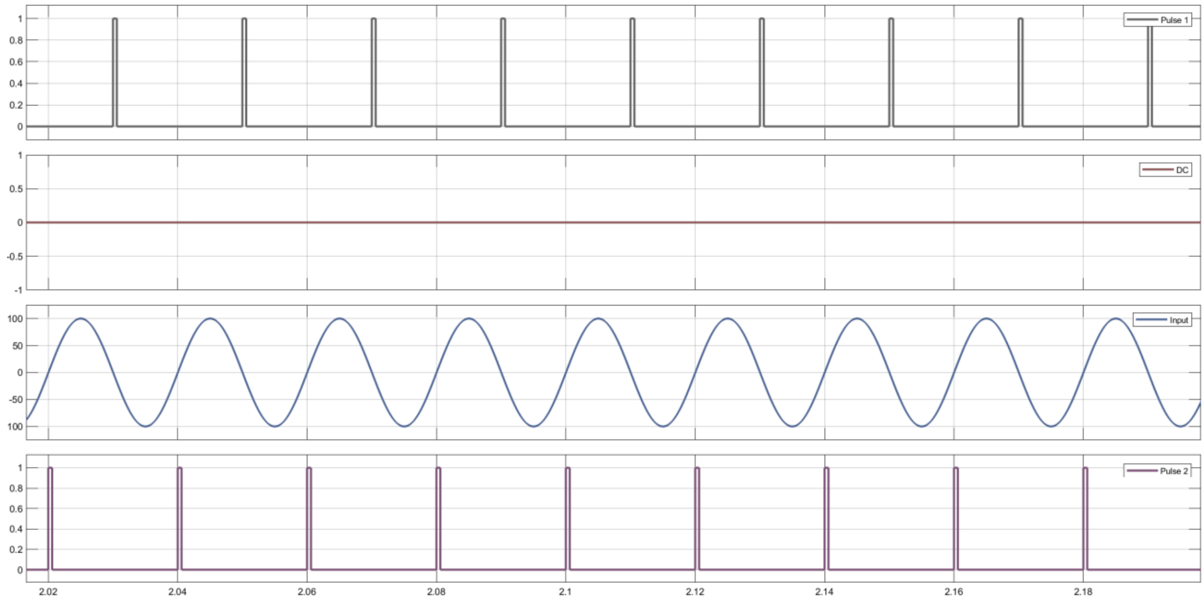


Fig. 3.2: Pulse1, DC, pulse2, and input waveform for controlled half- wave rectifier circuit

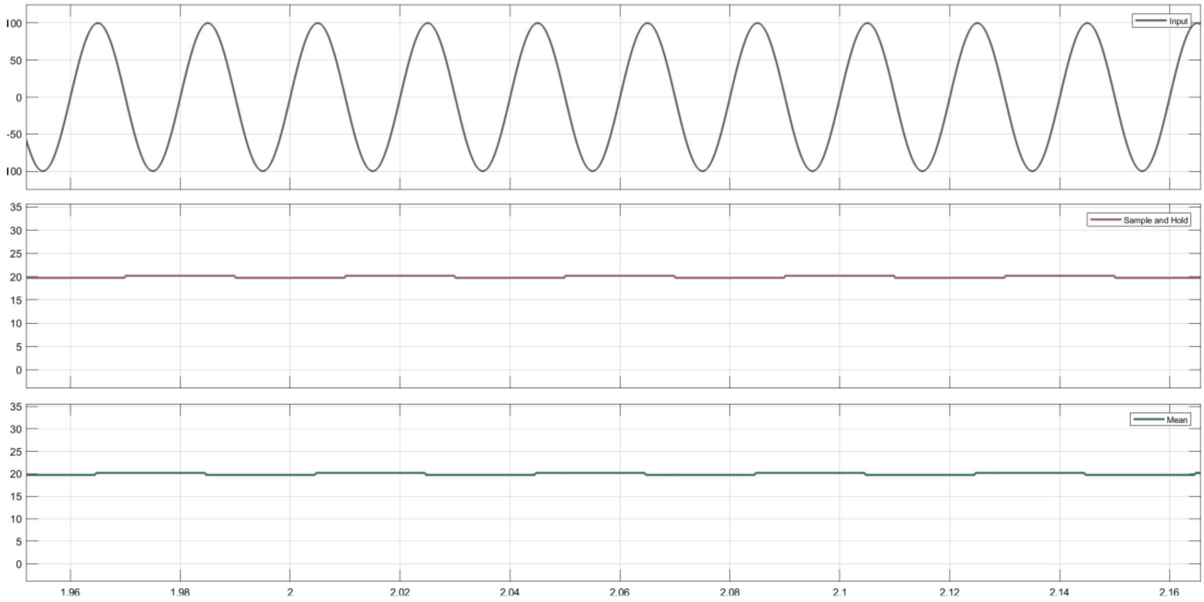


Fig. 3.3: Input, sample & hold, and mean waveform for controlled half- wave rectifier circuit

3.7 Discussion & Conclusion

In this experiment, triggered subsystem block was used. Additionally, to decrease the amount of block components in the Simulink platform, we employed MATLAB function blocks and codes. In the experiment, half-wave and full-wave controlled rectifier circuits are developed and studied.

Moreover, expected outputs were observed at scope. Thus, the experiment was a success.

Experiment No. 03

4.1 Experiment Name

AC voltage controller circuit and DC-DC converter circuit using Simulink

4.2 Objectives

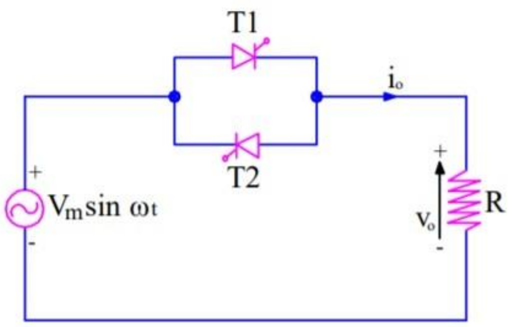
- To become acquainted with the Simulink platform and Simulink library
- To simulate various types of circuits using blocks from the library browser
- To design both AC voltage controller circuit and DC-DC converter circuit using Simulink

4.3 Theory

4.3.1 AC voltage controller circuit

AC Voltage Controller is a thyristor-based device which converts fixed alternating voltage directly to variable alternating voltage without a change in frequency. The working principle of AC Voltage Controller is based on either of two methods: Phase Control & Integral Cycle Control.

In the phase control method, the firing angle of the thyristor is used to manage the phase relationship between the commencement of the load current and the input supply voltage. In integral cycle control, the AC input supply is turned on for a number of integral cycles before being shut off again.



4.3.2 DC-DC converter (Boost Converter)

A direct current (DC) to direct current (DC) converter is an electrical circuit or electromechanical device that converts a direct current (DC) source from one voltage level to another. It is a form of energy converter. Power levels range from extremely low (tiny batteries) to extremely high (high-voltage power transmission).

DC-to-DC converters are widely used for DC microgrid applications at different voltage levels. They are designed to maximize the energy harvest from photovoltaic systems and wind turbines. The entire circuit is cheaper and more efficient than a simple mains transformer circuit with the same output.

4.4 Apparatus

- Simulink

4.5 Simulink Block Diagram & Waveform

- **AC voltage** controller circuit

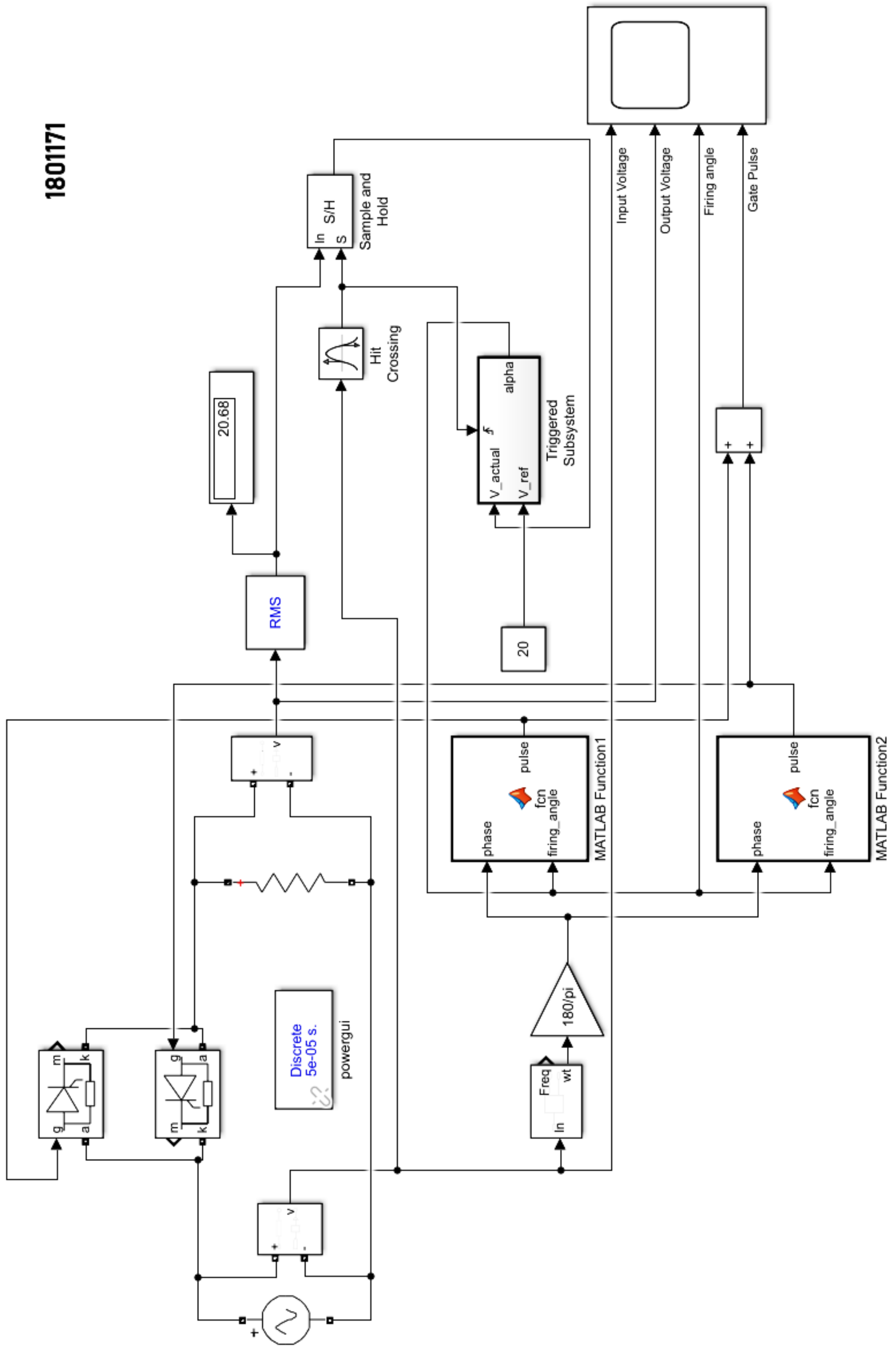


Fig. 4.1: Block diagram for AC voltage controller circuit

- **Code for triggered system**

```
function y = fcn(Vact,Vref)
step = 1.5;
%code_gen
persistent firing_angle
if isempty(firing_angle), firing_angle=10; end
if Vact<Vref
    firing_angle = firing_angle -step;
    if firing_angle<0
        firing_angle =0;
    end
elseif Vact>Vref
    firing_angle =firing_angle+step;
    if firing_angle>180
        firing_angle =180;
    end
else
    firing_angle =firing_angle;
end
y = firing_angle;
end
```

- **Code for MATLAB function (positive half cycle)**

```
function pulse = fcn(phase,firing_angle)
pulse_width = 10;

if (phase>=firing_angle)&& phase<=
min(firing_angle+pulse_width,180)
    pulse =1;
else
    pulse=0;
end
end
```

- **Code for MATLAB function (negative half cycle)**

```
function pulse = fcn(phase,firing_angle)
pulse_width = 10;

if (phase >=firing_angle+180)&& phase<=
min(firing_angle+pulse_width+180,360)
    pulse =1;
else
    pulse=0;
end
end
```

- **Waveform**

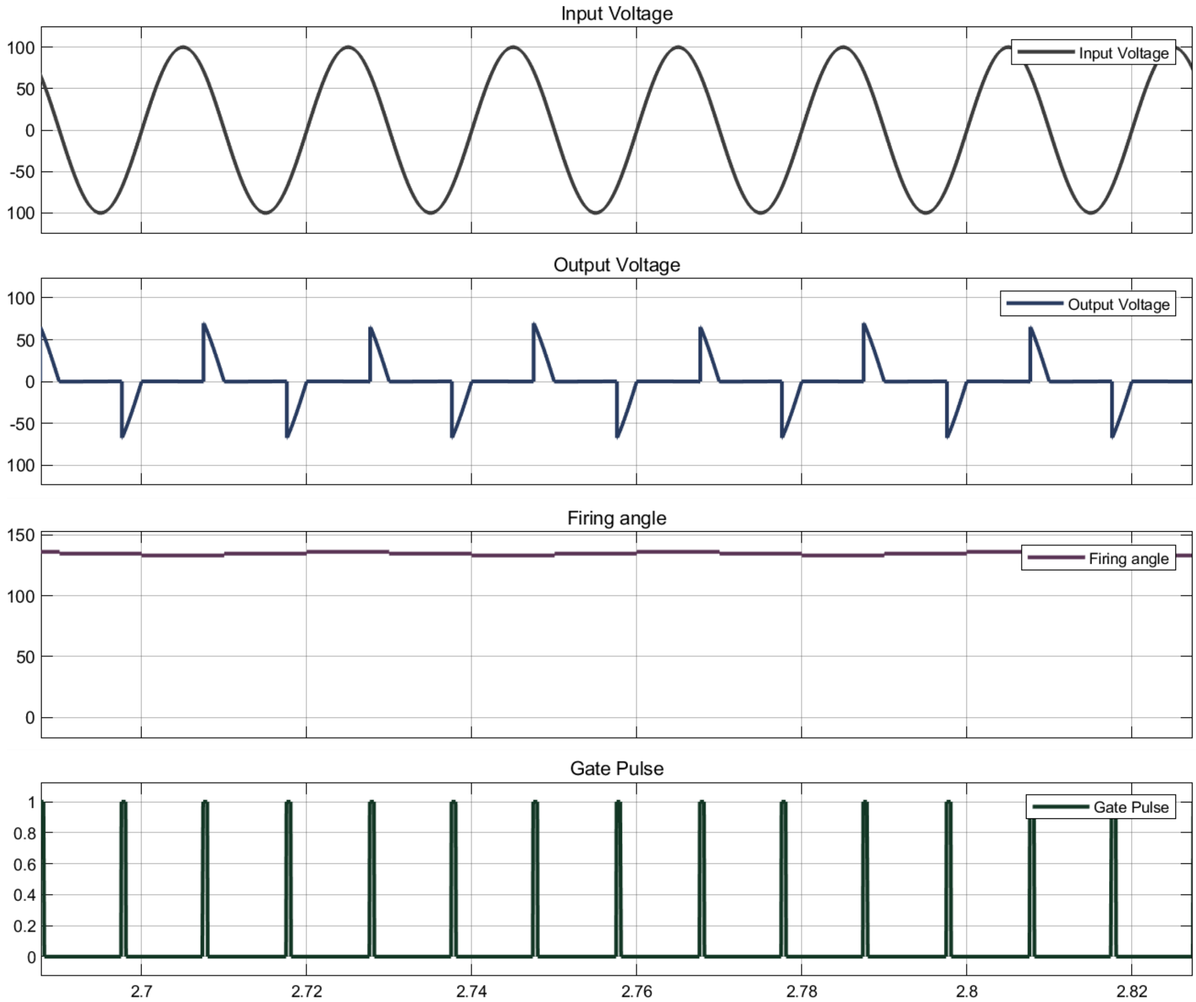


Fig. 4.2: Input, output, firing angle, and gate pulse waveform for AC voltage controller circuit

- **DC-DC converter (Boost Converter)**

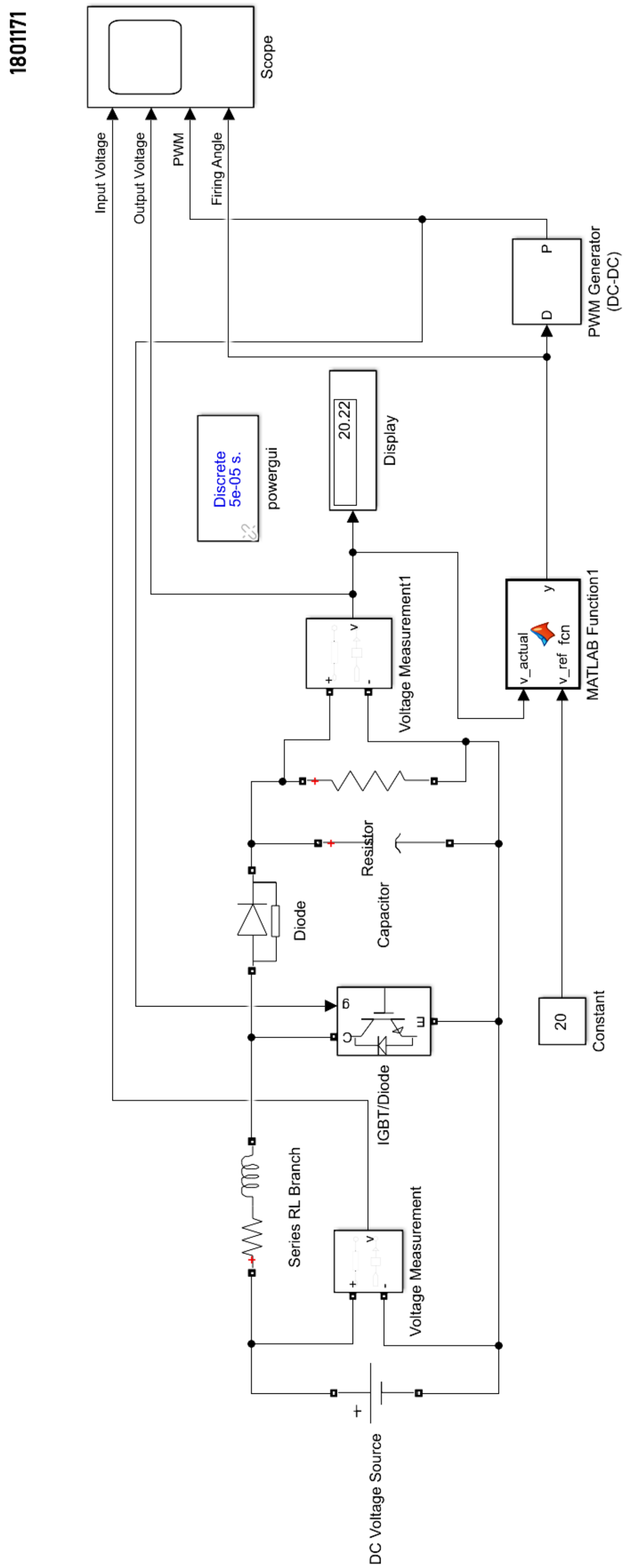


Fig. 4.3: Block diagram for DC-DC converter (Boost Converter)

- **Code for MATLAB function**

```
function y = fcn(v_actual,v_ref)
delta = 0.01
persistent D
if isempty(D), D = .1; end
if v_actual < v_ref
    D = D+delta;
    if D>=.75
        D = .75;
    end
elseif v_actual > v_ref
    D = D-delta;
    if D<0;
        D = 0;
    end
else
    D = D;
end
y=D;
end
```

- **Waveform**

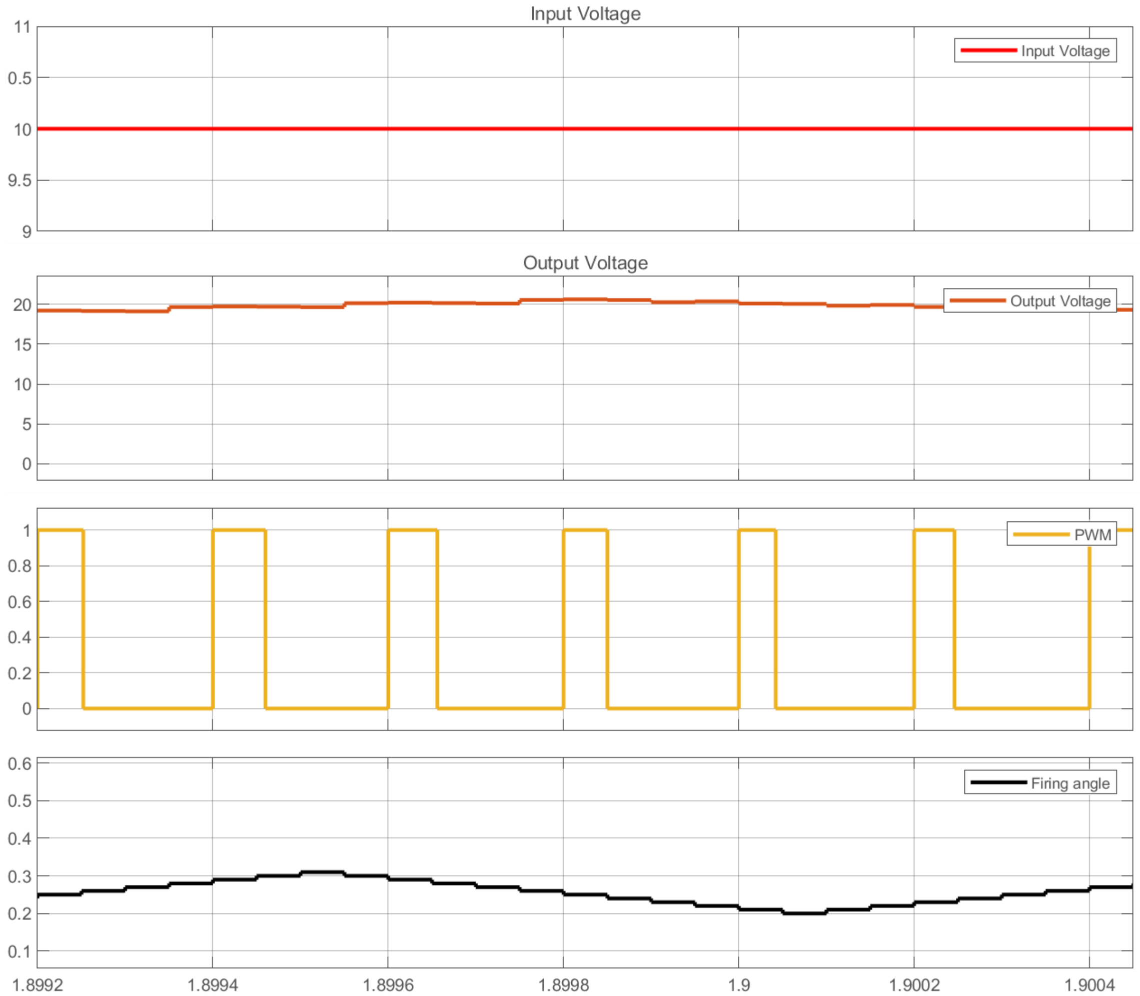


Fig. 4.4: Input, output, PWM, and firing angle waveform for DC-DC converter (Boost Converter)

4.6 Discussion & Conclusion

In this experiment, triggered subsystem block was used. Additionally, to decrease the amount of block components in the Simulink platform, we employed MATLAB function blocks and codes. We were able to successfully design an AC voltage regulator and a DC-DC converter (boost converter) circuit as a result of this experiment. Moreover, expected outputs were observed at scope. Thus, the experiment was a success.

4.7 Reference

- https://electricalbaba.com/what-is-ac-voltage-controller-definition-working-and-application/
- https://en.wikipedia.org/wiki/DC-to-DC_converter

Experiment No. 05

5.1 Experiment Name

DC-DC boost converter using pi controller and Single-phase H bridge DC-AC inverter using Simulink

5.2 Objectives

- To become acquainted with the Simulink platform and Simulink library
- To design and analyze DC-DC converter using pi controller
- To design and analyze Single-phase H bridge DC-AC inverter using Simulink platform

5.3 Theory

5.3.1 DC-DC boost converter

A direct current (DC) to direct current (DC) converter is an electrical circuit or electromechanical device that converts a direct current (DC) source from one voltage level to another. It is a form of energy converter. Power levels range from extremely low (tiny batteries) to extremely high (high-voltage power transmission).

5.3.2 Single-phase H bridge DC-AC inverter

The inverter is a device that converts a dc voltage into ac voltage and it consists of four switches. It is also known as full-bridge inverter circuit. A full-bridge inverter's circuit consists of four diodes and four regulated switches. Because they feed the stored energy in the load back into the DC source, these diodes are known as freewheeling diodes.

5.4 Apparatus

- Simulink

5.5 Simulink Block Diagram & Waveform

- **DC-DC boost converter using pi controller**
- **Code for MATLAB function**

```
function y = fcn(Vact,Vref)
ss=0.05;
persistent d
if isempty(d),d=0; end
if Vact<Vref
      d=d+ss;
      if d>0.65
         d=0.65;
      end
elseif Vact>Vref
      d=d-ss;
      if d<0
         d=0;
      end
      else
         d=d;
       end
y=d;
end
```

- **Block diagram**

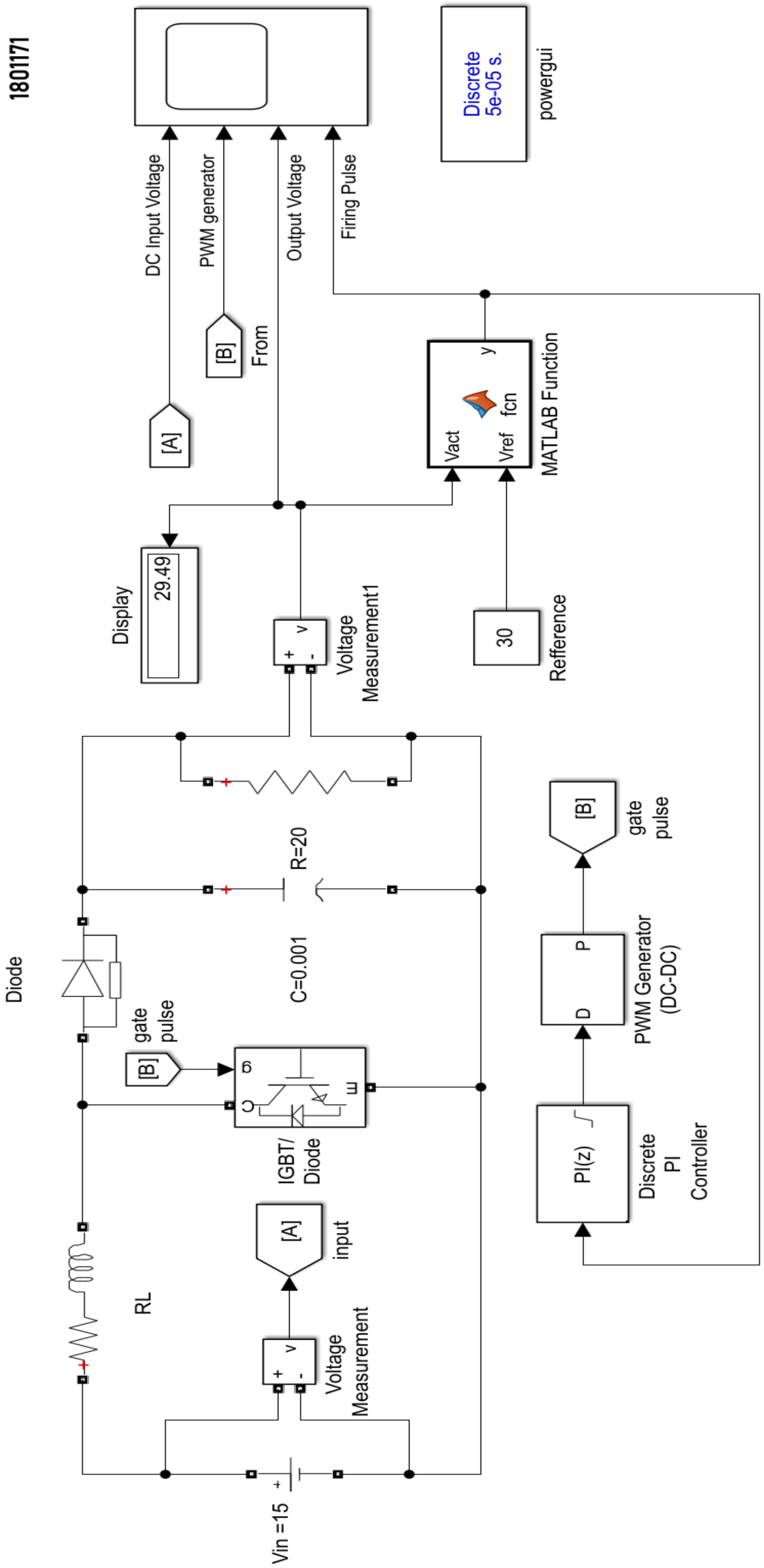


Fig. 5.1: Block diagram for DC-DC boost converter using pi controller

- **Waveform**

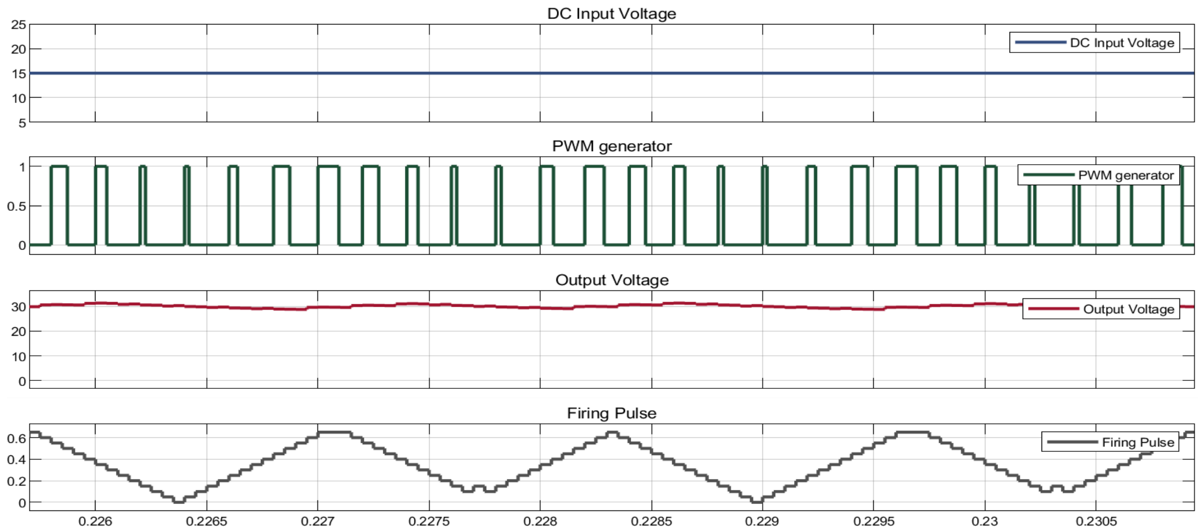


Fig. 5.2: Waveform for DC-DC boost converter using pi controller

- **Single-phase H bridge DC-AC inverter**
- **Code for MATLAB function**

```
function [Q1 ,Q2 ,Q3 ,Q4,Vc1,Vc2] = pulse(Vref, Vt)
Vc1 = (Vt+1)*0.5;
Vc2 = (Vt-1)*0.5;
if Vref>0
    if Vref >= Vc1
        Q1=1;
        Q2=1;
        Q3=0;
        Q4=0;
    else
        Q1=1;
        Q2=0;
        Q3=0;
        Q4=1;
    end
elseif Vref<0
    if Vref<=Vc2
        Q1=0;
        Q2=0;
        Q3=1;
        Q4=1;
    else
        Q1=0;
        Q2=1;
        Q3=1;
        Q4=0;
    end
else
    Q1=0;
    Q2=0;
    Q3=0;
    Q4=0;
end
end
```

- **Block diagram**

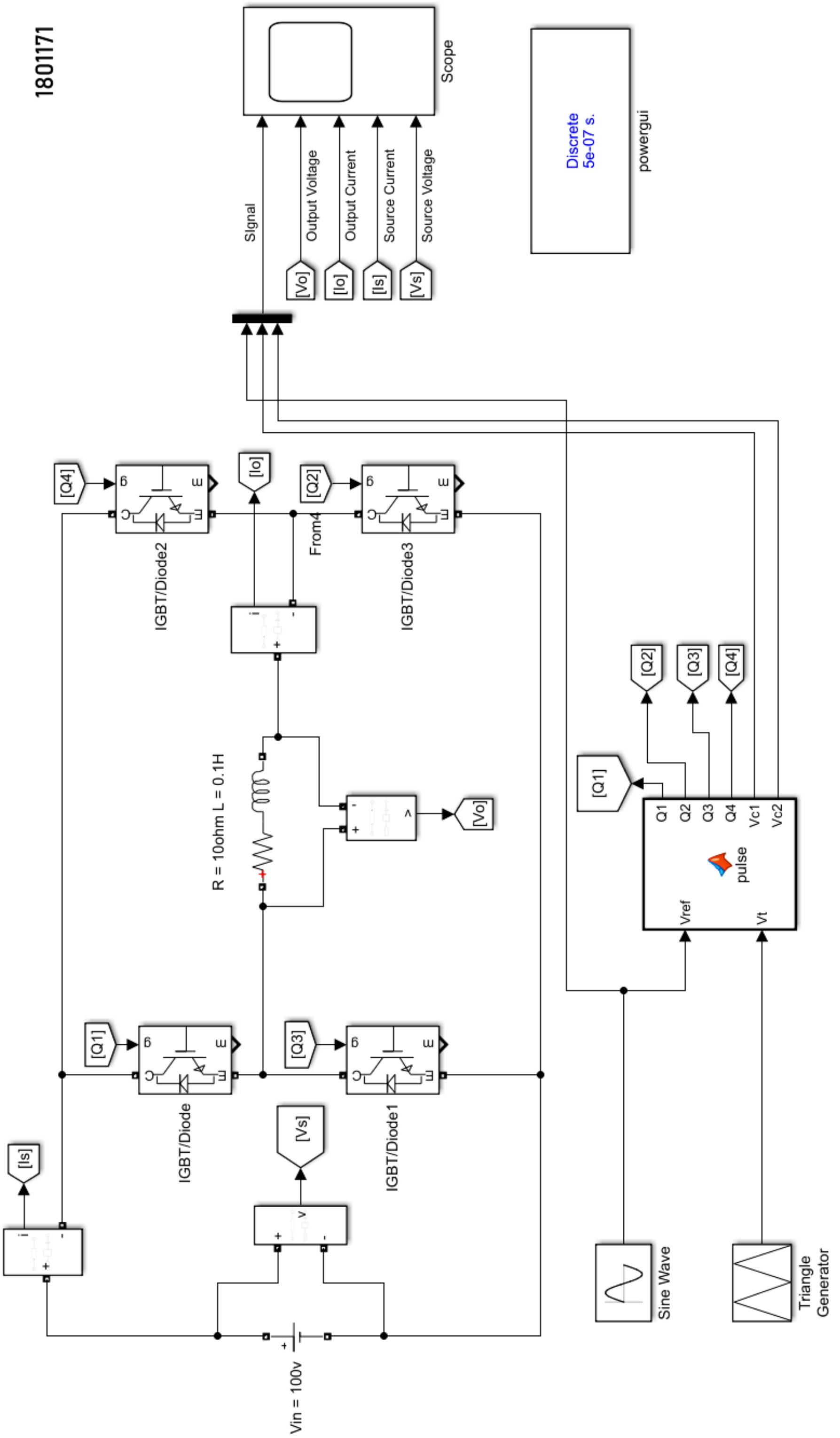


Fig. 5.5: Block diagram for Single-phase H bridge DC to AC inverter

- **Waveform**

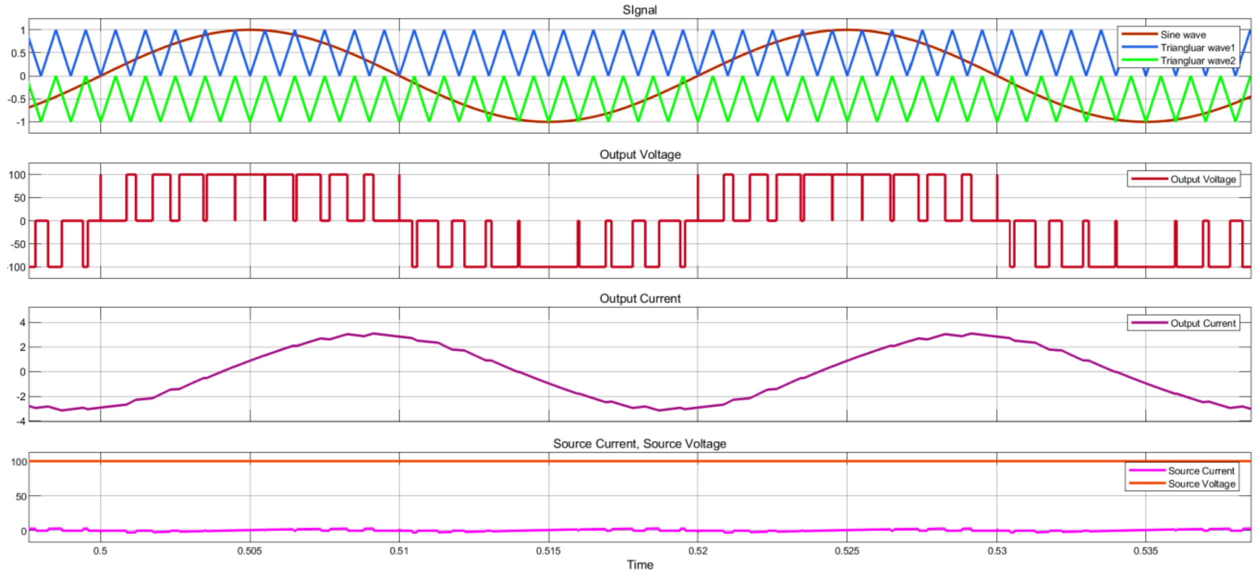


Fig. 5.4: Waveform for Single-phase H bridge DC to AC inverter

5.6 Discussion & Conclusion

In this experiment, we were able to successfully design a DC-DC boost converter circuit using a Pi controller and a single-phase H-bridge DC-AC inverter using Simulink. We used IGBT and MATLAB functions and analyzed their characteristics through this experiment. Moreover, we compared our theoretically obtained waveform with the simulated waveform. In the end, expected outputs were observed within scope. Thus, the experiment was a success.

Experiment No. 06

6.1 Experiment Name

Cascaded Single Phase and Three Phase H-Bridge Inverter using Simulink

6.2 Objectives

- To get familiarize with the Simulink platform and Simulink library
- To develop and study a cascaded single-phase H-bridge inverter using Simulink
- To use the Simulink platform to construct and analyze a cascaded three-phase H-bridge inverter.

6.3 Theory

Single phase and three phase H-bridge inverter

The inverter is a device that converts dc voltage to alternating current voltage and consists of four switches, whereas the half-bridge inverter requires two diodes and two switches connected in parallel. This can be constructed and cascaded into single and three phase inverters as desired.

The primary distinction between a single-phase and three-phase inverter is that a single-phase inverter can generate single-phase power from PV modules. It can also connect to single-phase equipment or the power grid. A three-phase, on the other hand, turns the DC input of solar panels into a three-phase AC output.

6.4 Apparatus

- Simulink

6.5 Simulink Block Diagram & Waveform

Cascaded Single-Phase H-bridge inverter

- **Block diagram**

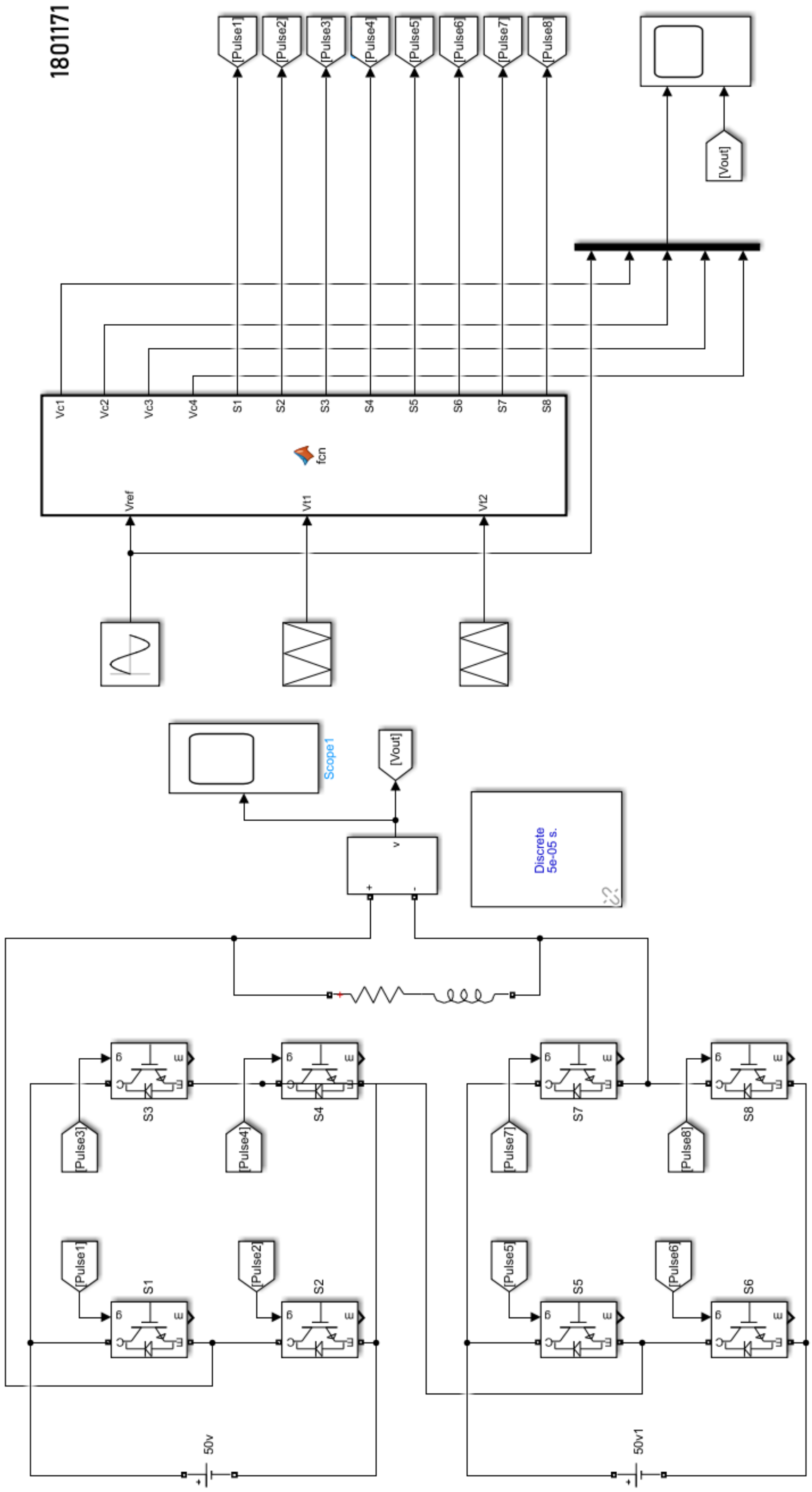


Fig. 6.1: Block diagram for Cascaded Single-Phase H-bridge inverter

- **Code for MATLAB function**

```
function [Vc1, Vc2, Vc3, Vc4, S1, S2, S3, S4, S5, S6, S7, S8] = fcn(Vref,
Vt1, Vt2)
Vc1 = (Vt1+1)*.25
Vc2 = .5+((Vt1+1)*.25)
Vc3 = (Vt2-1)*.25
Vc4 = -0.5+((Vt2-1)*.25)
if Vref>=0
if Vref>=0 && Vref<=0.5
    if  Vref>=Vc1
        S1=1; S2=0; S3=0; S4=1; S5=1; S6=0; S7=1; S8=0;
    else
        S1=1; S2=0; S3=1; S4=0; S5=1; S6=0; S7=1; S8=0;
    end
else Vref>0.5 && Vref<=1
     if  Vref>=Vc2
        S1=1; S2=0; S3=0; S4=1; S5=1; S6=0; S7=0; S8=1;
    else
       S1=1; S2=0; S3=0; S4=1; S5=1; S6=0; S7=1; S8=0;
     end
end

else
if Vref<0 && Vref>=-0.5
    if  Vref<=Vc3
        S1=0; S2=1; S3=1; S4=0; S5=1; S6=0; S7=1; S8=0;
    else
        S1=1; S2=0; S3=1; S4=0; S5=1; S6=0; S7=1; S8=0;
    end
else Vref<-0.5 && Vref>=-1
     if  Vref<=Vc4
        S1=0; S2=1; S3=1; S4=0; S5=0; S6=1; S7=1; S8=0;
    else
       S1=0; S2=1; S3=1; S4=0; S5=1; S6=0; S7=1; S8=0;
     end
end
end
end
```

- **Waveform**

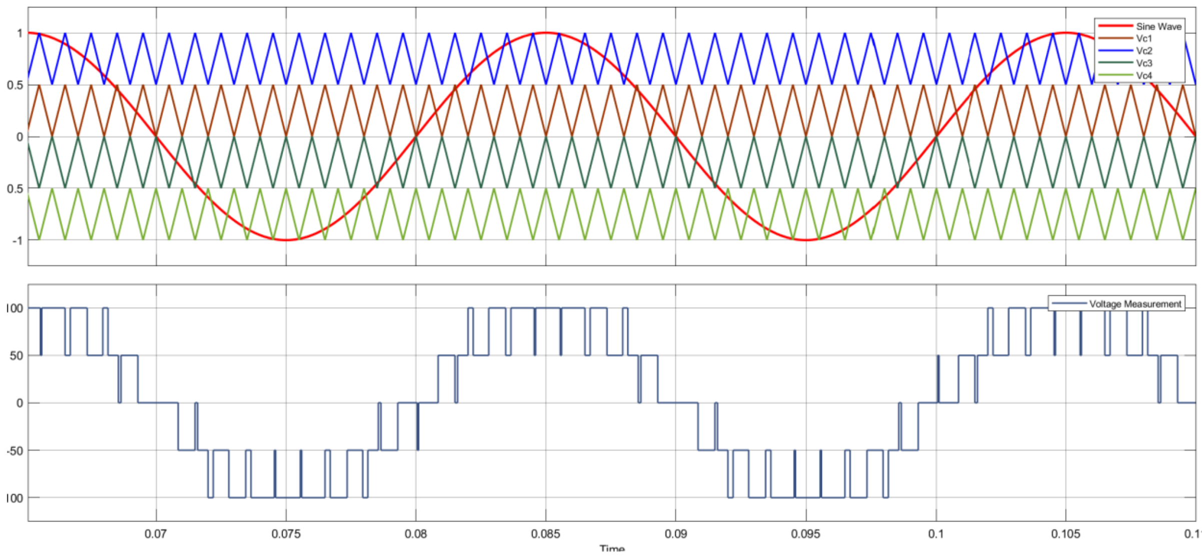


Fig. 6.2: Waveform for Cascaded Single-Phase H-bridge inverter

Three-phase H bridge inverter

- **Code for MATLAB function1**

```
function [Vc1,Vc2,Vc3,Vc4] = fcn(Vc)
Vc1 = (Vc+1)/4;
Vc2=Vc1+0.5;
Vc3= (Vc+1)/(-4);
Vc4=Vc3-0.5;
end
```

- **Code for MATLAB function2**

```
function [S1,S2,S3,S4,S5,S6,S7,S8]  = fcn(Vref,Vc1,Vc2,Vc3,Vc4)

if Vref >0
    if Vref>=0 && Vref<=0.5
        if Vref>=Vc1
            S1 =1; S2 =0; S3 =0; S4 =1; S5 =1; S6 =0; S7 =1; S8 =0;
        else
            S1 =1; S2 =0; S3 =1; S4 =0; S5 =1; S6 =0; S7 =1; S8 =0;
        end

    else
        if Vref>=Vc2
            S1 =1; S2 =0; S3 =0; S4 =1; S5 =1; S6 =0; S7 =0; S8 =1;
        else
            S1 =1; S2 =0; S3 =0; S4 =1; S5 =1; S6 =0; S7 =1; S8 =0;
        end
    end

else
    if Vref<=0 && Vref>=-0.5
        if Vref<=Vc3
            S1 =0; S2 =1; S3 =1; S4 =0; S5 =1; S6 =0; S7 =1; S8 =0;
        else
            S1 =0; S2 =1; S3 =0; S4 =1; S5 =0; S6 =1; S7 =0; S8 =1;
        end

    else
        if Vref<=Vc4
            S1 =0; S2 =1; S3 =1; S4 =0; S5 =0; S6 =1; S7 =1; S8 =0;
        else
            S1 =0; S2 =1; S3 =1; S4 =0; S5 =1; S6 =0; S7 =1; S8 =0;
        end
    end
end
end
```

- **Block diagram**

1801171

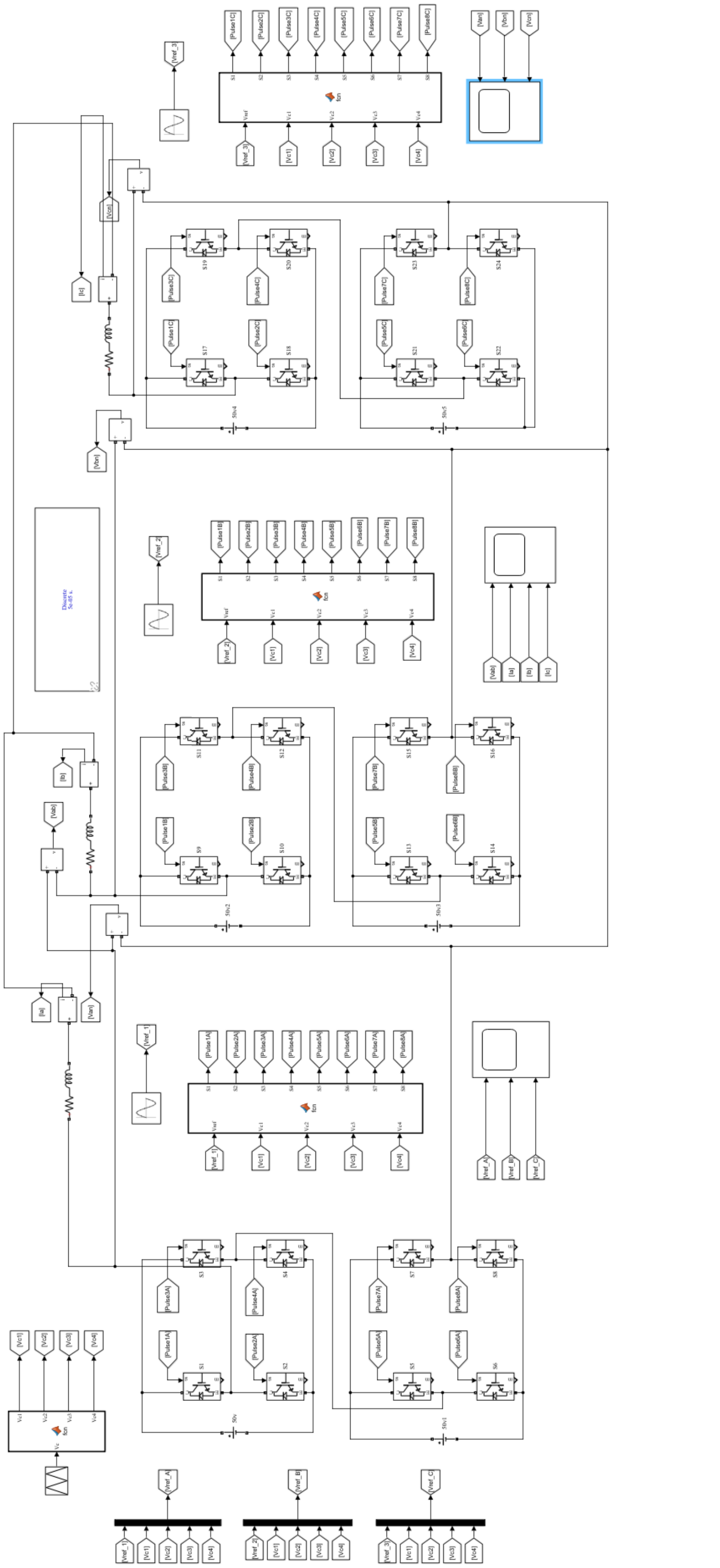
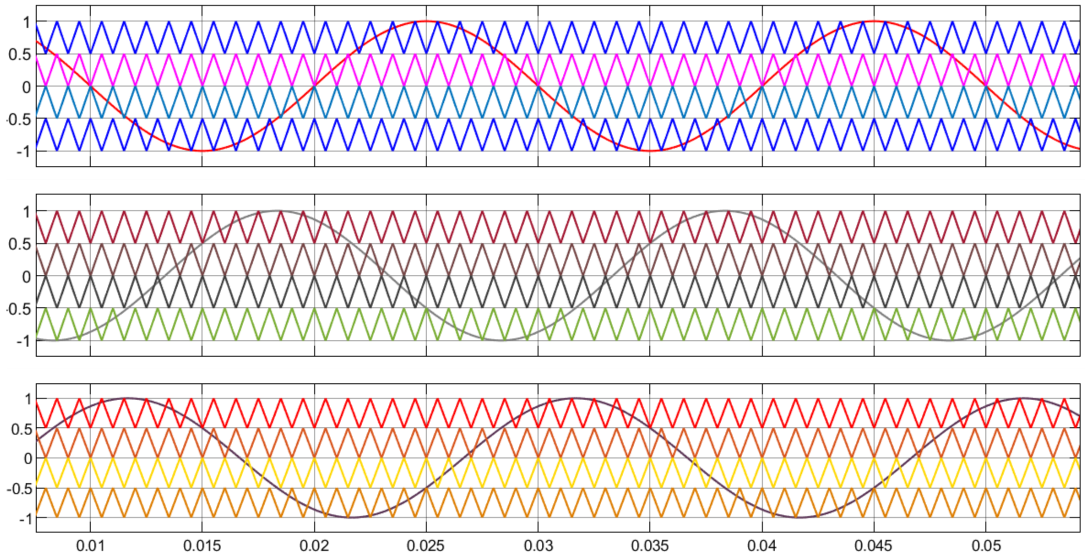
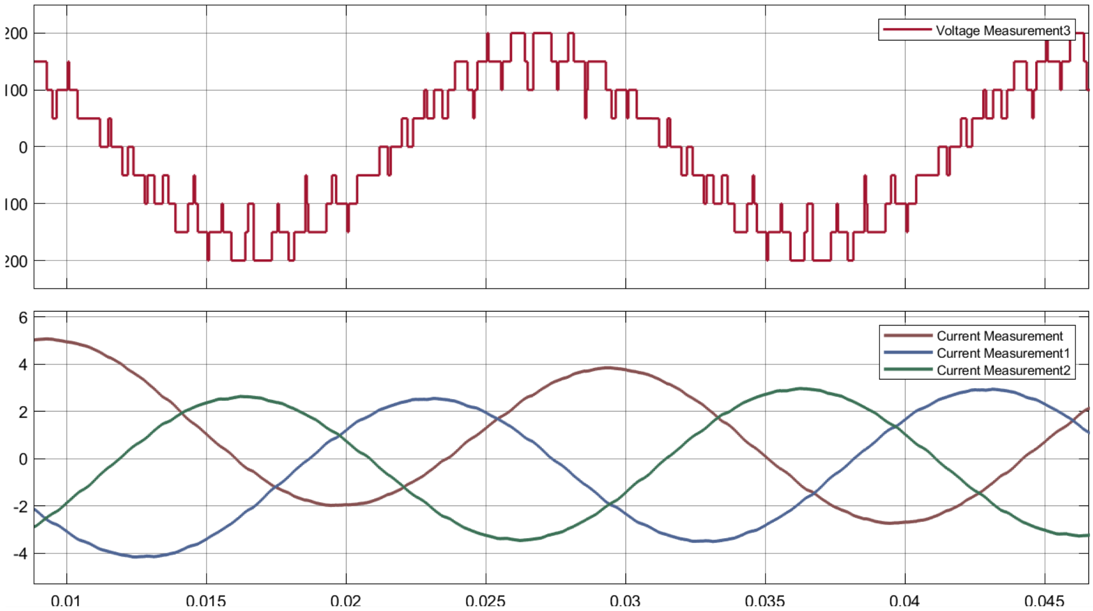


Fig. 6.3: Block diagram for Cascaded Single-Phase H-bridge inverter

- **Waveform**


0.01
0.015
0.02
0.025
0.03
0.035
0.04
0.045
0.05


Voltage Measurement3
Current Measurement
Current Measurement1
Current Measurement2
0.01
0.015
0.02
0.025
0.03
0.035
0.04
0.045

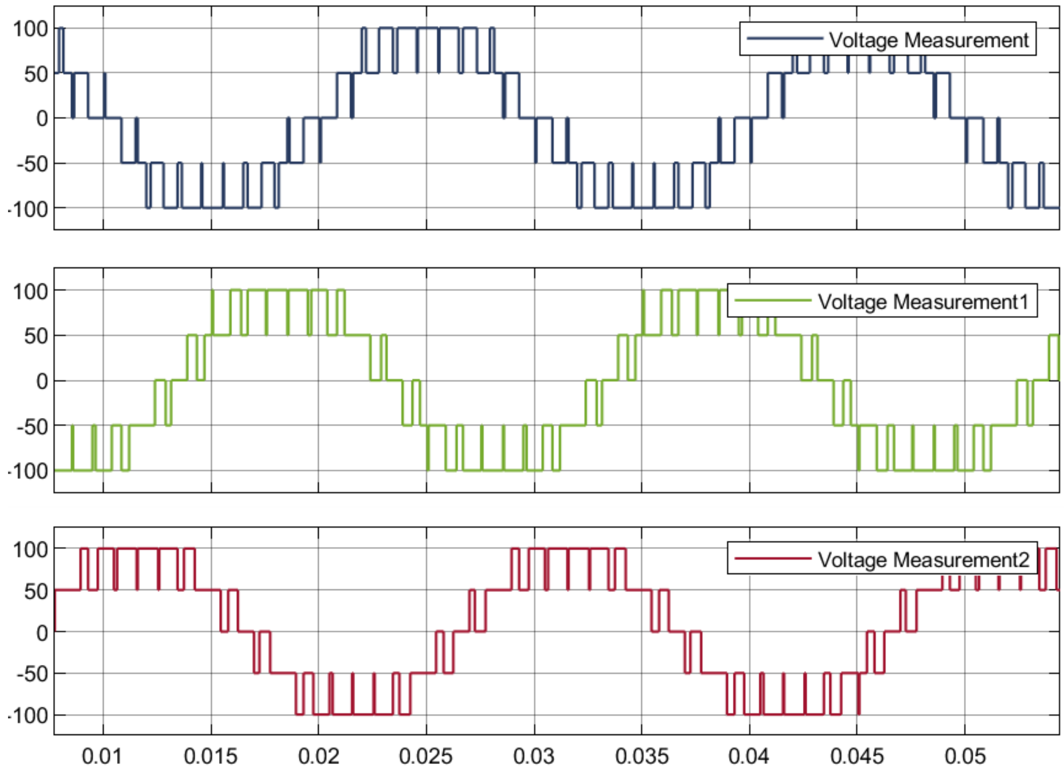


Fig. 6.4: Waveshape of PWM generators (Reference Voltages- Vref_A, Vref_B & Vref_C) , Phase Voltages (Van, Vbn & Vcn), Line Voltage (Vab) and Line Current (Ia) of Cascaded Three Phase H-Bridge Inverter Circuit

6.6 Discussion & Conclusion

This experiment thoroughly investigated cascaded single-phase and three-phase H-bridge inverters. RL load open loop was used to design single-phase and three-phase cascaded H-bridge inverter circuits. Four triangular waves of varying amplitudes were used to generate reference voltages. Three single-phase sources of equal amplitude shared a 120° phase shift in a three-phase cascaded inverter circuit. Finally, the experiment was carried out step by step, with each step carried out with honesty and care. As a result, the experiment is said to be a success.

Experiment No. 07

7.1 Experiment Name

Three phase inverter and hysteresis control of grid connected single phase inverter using Simulink

7.2 Objectives

- To get familiarize with the Simulink platform and Simulink library
- To develop and study a Three phase inverter using Simulink
- To use the Simulink platform to construct and analyze a hysteresis control of grid connected single phase inverter

7.3 Theory

Three phase inverters

The inverter is a device that converts dc voltage to alternating current voltage and consists of four switches, whereas the half-bridge inverter requires two diodes and two switches connected in parallel. This can be constructed and cascaded into single and three phase inverters as desired.

The primary distinction between a single-phase and three-phase inverter is that a single-phase inverter can generate single-phase power from PV modules. It can also connect to single-phase equipment or the power grid. A three-phase, on the other hand, turns the DC input of solar panels into a three-phase AC output.

Hysteresis control of grid connected single phase inverter

The purpose of the current controller is to control the load current by forcing it to follow a reference one. It is achieved by the switching action of the inverter to keep the current within the hysteresis band. The main advantages of this nonlinear control technique are related to its simple implementation, performing time response, and robustness.

7.4 Apparatus

- Simulink

7.5 Simulink Block Diagram & Waveform

Three phase inverters

- **Code for MATLAB function**

```
function [g1,g2,g3,g4, g5,g6] = fcn(Va, Vb, Vc,Vca)
 if Va>=Vca
    g1=1;    g2=0;
else
    g1=0;    g2=1;
end
if Vb>=Vca
    g3=1;    g4=0;
else
    g3=0;    g4=1;
end
if Vc>=Vca
    g5=1;    g6=0;
else
    g5=0;    g6=1;
end
end
```

- **Block diagram**

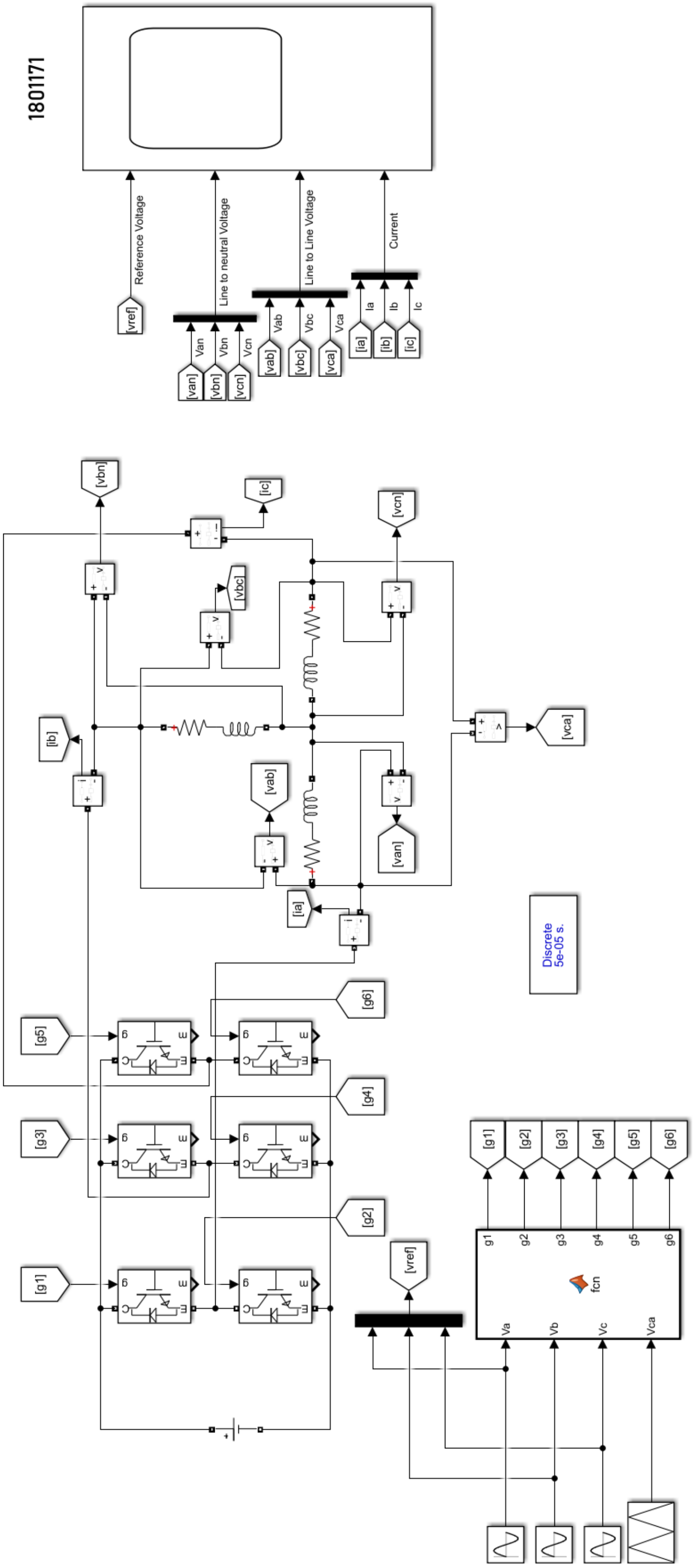


Fig. 7.1: Block diagram for Three-Phase inverter

- **Waveform**

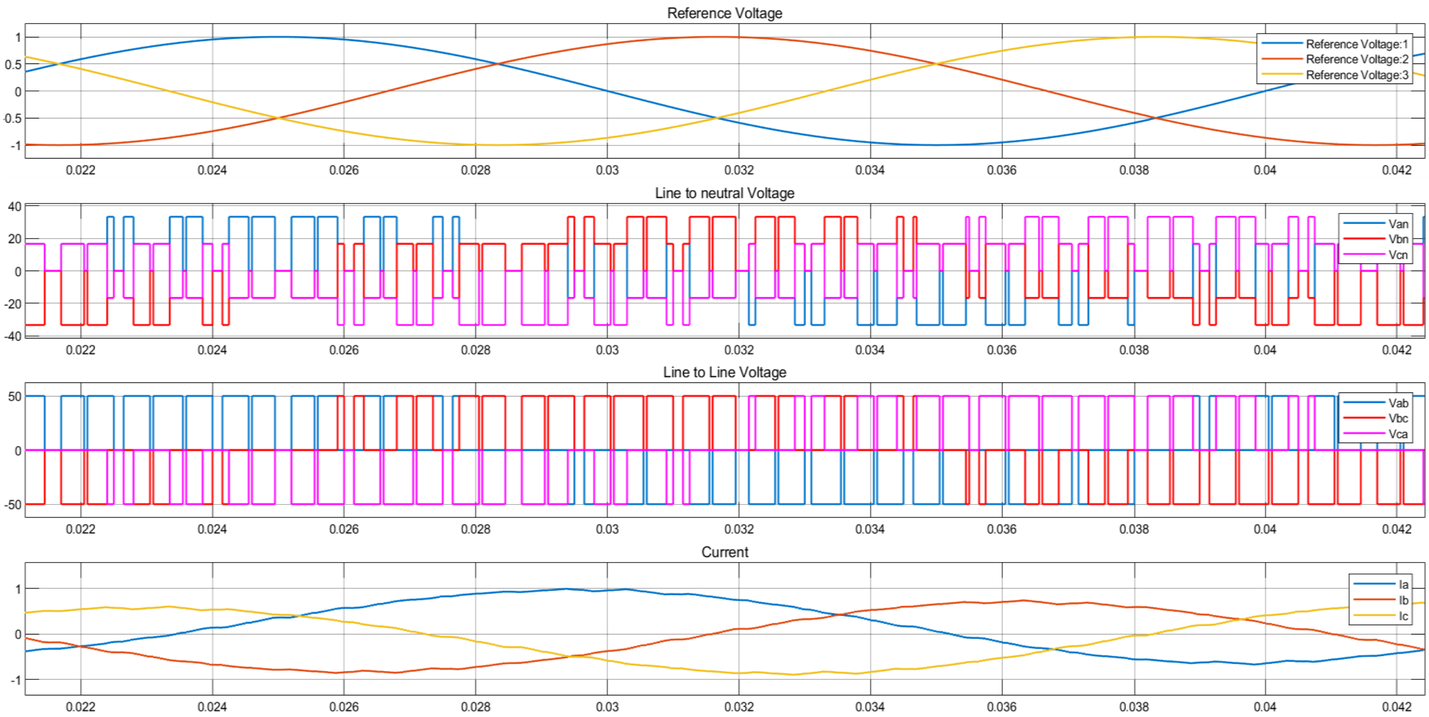


Fig. 7.2: Waveform for Three-Phase inverter

Hysteresis control of grid connected single phase inverter

- **Block diagram**

1801171

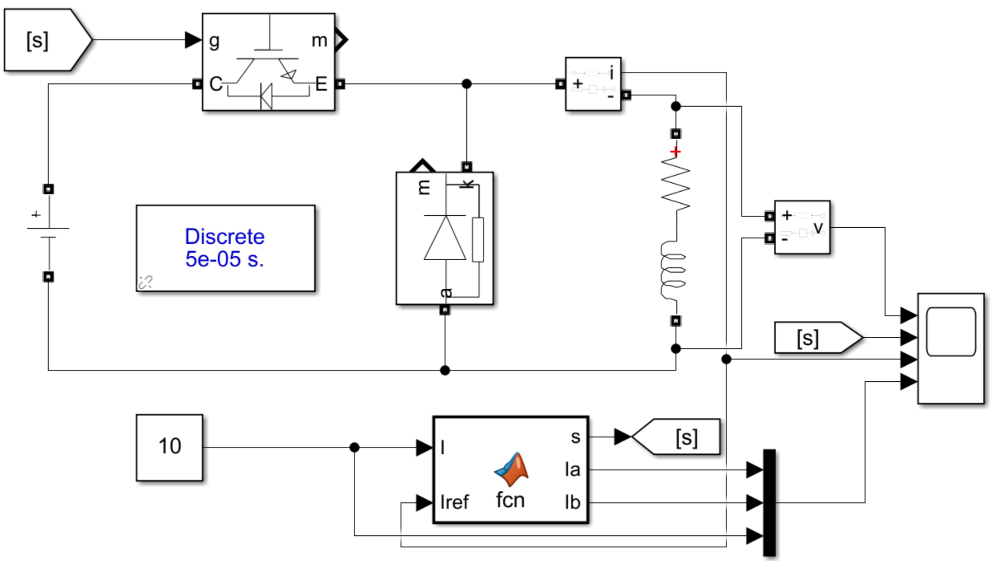


Fig. 7.3: Block diagram for hysteresis control of grid connected single phase inverter

- **Code for MATLAB function**

```
function [s,Ia,Ib]  = fcn(I,Iref)
persistent d
if isempty(d),d=1;end
e=1
Ia=I+e;
Ib=I-e;
```


1801171

```
if Iref>Ia
    d =0;
elseif Iref<Ib
    d=1;
else
    d=d;
end
s=d
end
```

- **Waveform**

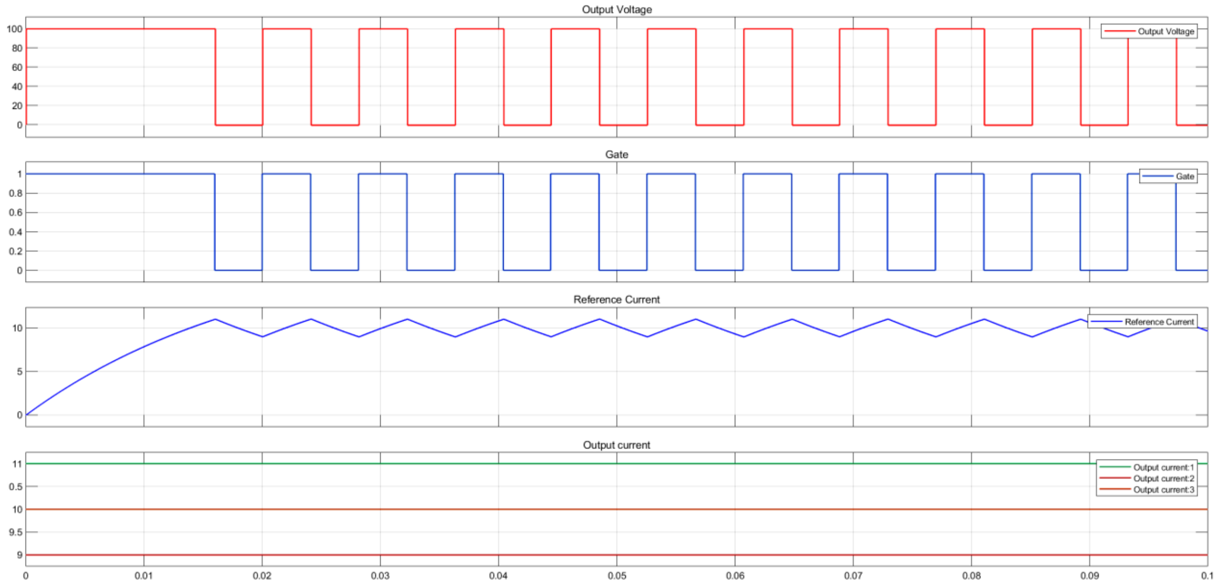


Fig. 7.2: Waveform for hysteresis control of grid connected single phase inverter

7.6 Discussion & Conclusion

This experiment thoroughly investigated three-phase and hysteresis control of grid connected single phase inverter. For three phase connection, we utilized Wy-delta connection according to our preference. Similarly for Hysteresis control inverter we used necessary tool according to our requirements. Thus, desired output was observed and the simulation was a success.

Experiment No. 08

8.1 Experiment Name

Simulation on hysteresis control of grid connected H-bridge system

8.2 Objectives

- To get familiarize with the Simulink platform and Simulink library
- To develop and study a H bridge inverter using Simulink
- To use the Simulink platform to construct and analyze a hysteresis control of H bridge inverter

8.3 Theory

Hysteresis control of grid connected H-bridge system

The purpose of the current controller is to control the load current by forcing it to follow a reference one. It is achieved by the switching action of the inverter to keep the current within the hysteresis band. The main advantages of this nonlinear control technique are related to its simple implementation, performing time response, and robustness.

8.4 Apparatus

- Simulink

8.5 Simulink Block Diagram & Waveform

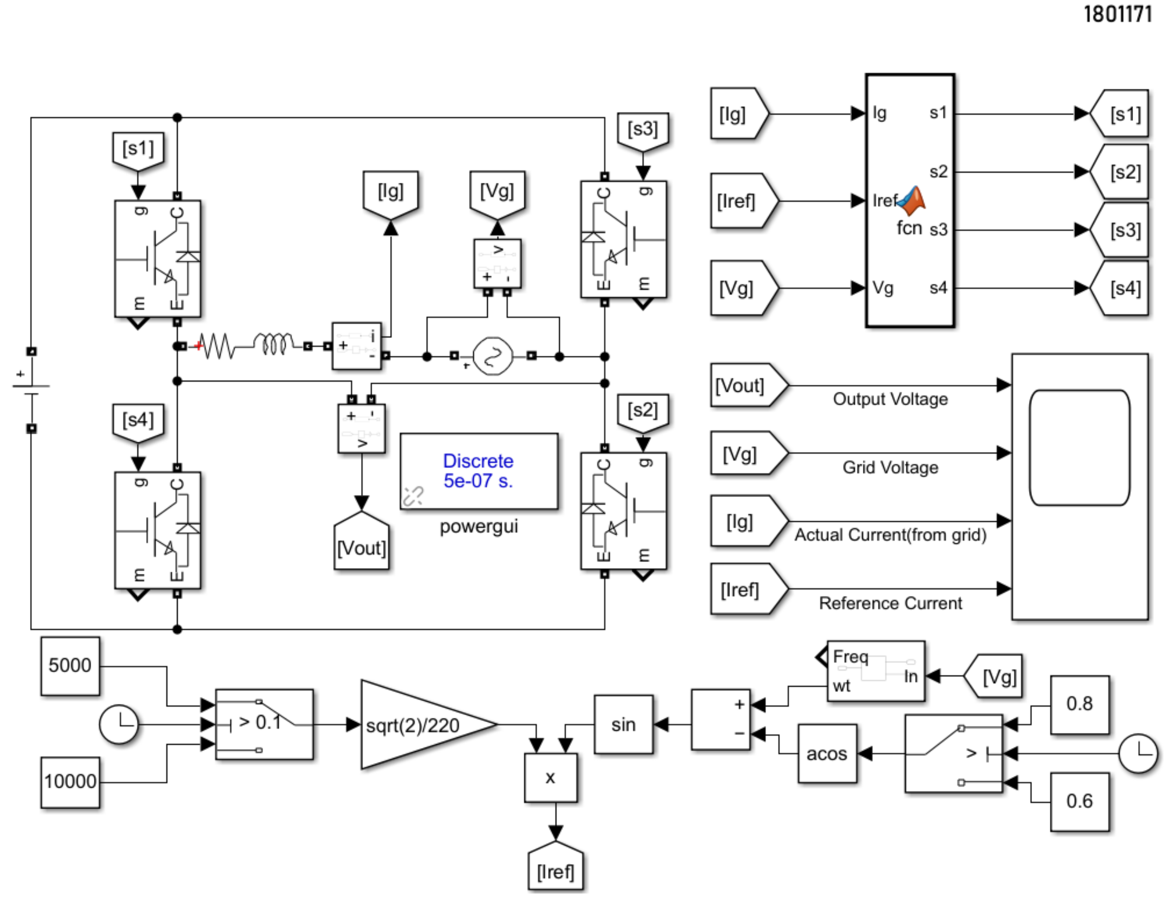


Fig.8.1: Block diagram of hysteresis control of grid connected H bridge system

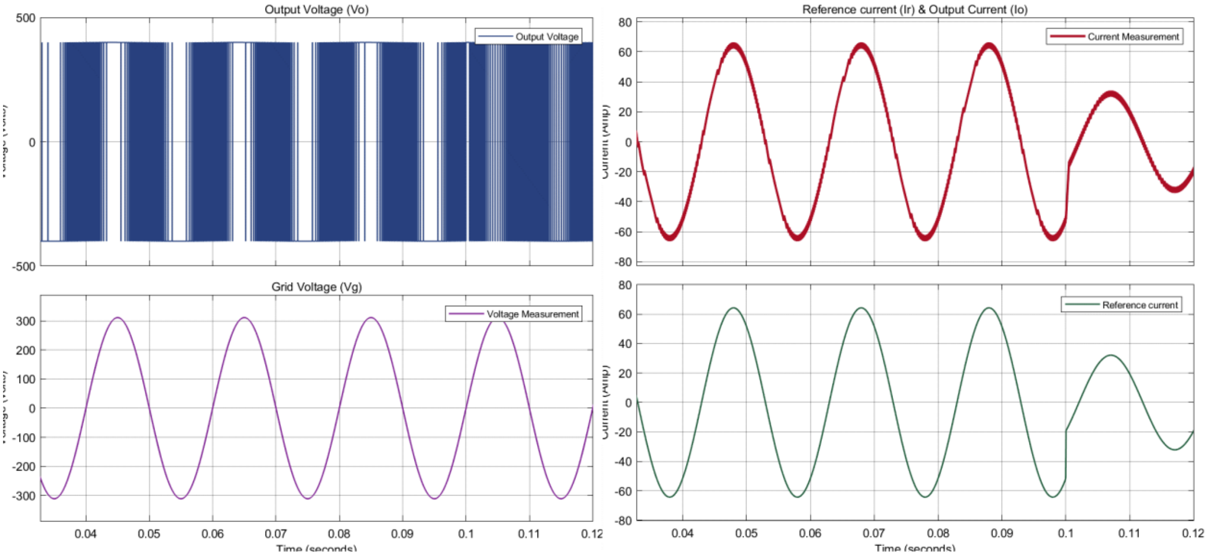


Fig.8.2: Waveform of bipolar hysteresis control of grid connected H bridge system

MATLAB Code

```
function [s1,s2,S3,S4] = fcn(Vg,Ig,Iref)
persistent Q1; if isempty(Q1)     Q1=0; end
persistent Q2; if isempty(Q2)     Q2=0; end
persistent Q3; if isempty(Q3)     Q3=0; end
persistent Q4; if isempty(Q4)     Q4=0; end
delta=1;
e=Ig-Iref;
if Vg>=0
    if e<=-delta
        Q1=1;   Q2=1;   Q3=0;   Q4=0;
    elseif e>=delta
        Q1=1;   Q2=0;   Q3=1;   Q4=0;
    else
        Q1=Q1;  Q2=Q2;  Q3=Q3;  Q4=Q4;
    end
else
    if e<=-delta
        Q1=1;   Q2=0;   Q3=1;   Q4=0;
    elseif e>=delta
        Q1=0;   Q2=0;   Q3=1;   Q4=1;
    else
        Q1=Q1;  Q2=Q2;  Q3=Q3;  Q4=Q4;
    end
   end
   s1=Q1;    s2=Q2;    S3=Q3;    S4=Q4;
end
```

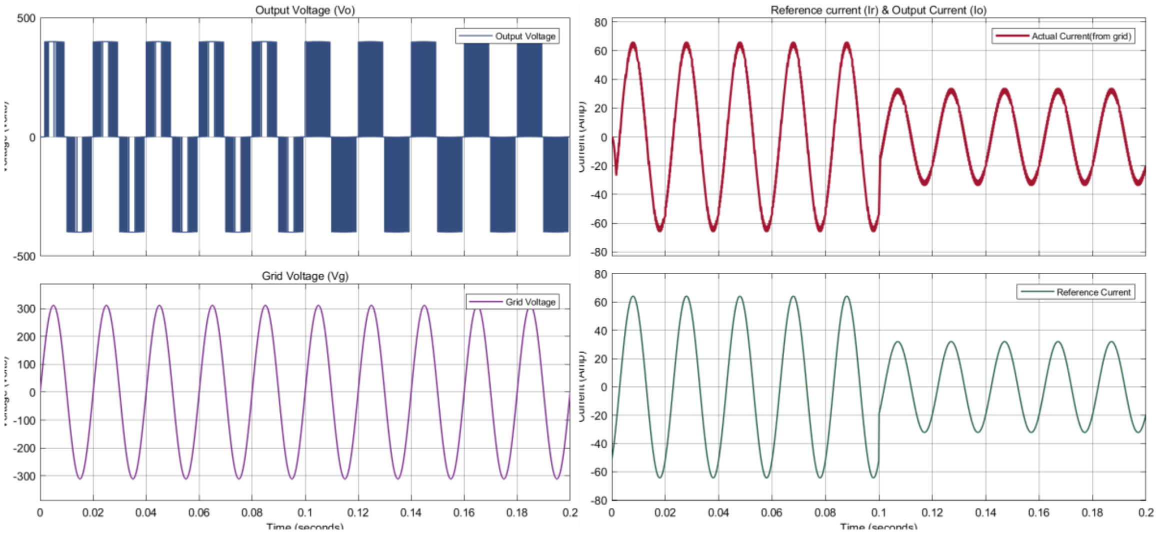


Fig.8.2: Waveform of unipolar hysteresis control of grid connected H bridge system

MATLAB Code

```
function [s1,s2,s3,s4] = fcn(Ig,Iref,Vg)
persistent Q1
if isempty(Q1);Q1=1;end
persistent Q2
if isempty(Q2);Q2=1;end
persistent Q3
if isempty(Q3);Q3=0;end
persistent Q4
if isempty(Q4);Q4=0;end
hd=2;
e=Ig-Iref;
if Vg>=0
    if e<=-hd
        Q1=1;Q2=1;Q3=0;Q4=0;
    elseif e>=hd
        Q1=1;Q2=0;Q3=1;Q4=0;
    else
        Q1 = Q1;Q2=Q2;Q3=Q3;Q4=Q4;
    end
else
    if e<=-hd
        Q1=0;Q2=1;Q3=0;Q4=1;
    elseif e>=hd
        Q1=0;Q2=0;Q3=1;Q4=1;
    else
        Q1 = Q1;Q2=Q2;Q3=Q3;Q4=Q4;
    end
end
s1=Q1; s2=Q2; s3=Q3; s4=Q4;
end
```

8.6 Discussion & Conclusion

This experiment thoroughly investigated hysteresis control of grid connected H bridge system. For this system, we utilized connection and value of parameter according to our preference. Similarly, for Hysteresis control inverter, we used necessary tool according to our requirements. Thus, desired output was observed and the simulation was a success.

Experiment No. 09

9.1 Experiment Name
Simulation on bi-directional hysteresis control of grid connected H- bridge and battery with controlled DC linked voltage

9.2 Objectives
- To develop and study a H bridge inverter using Simulink
- To get familiarize with the Simulink platform and Simulink library
- To use the Simulink platform to construct and analyze a bi-directional hysteresis control of grid connected H bridge inverter

9.3 Apparatus
- Simulink

9.4 Simulink Block Diagram & Waveform

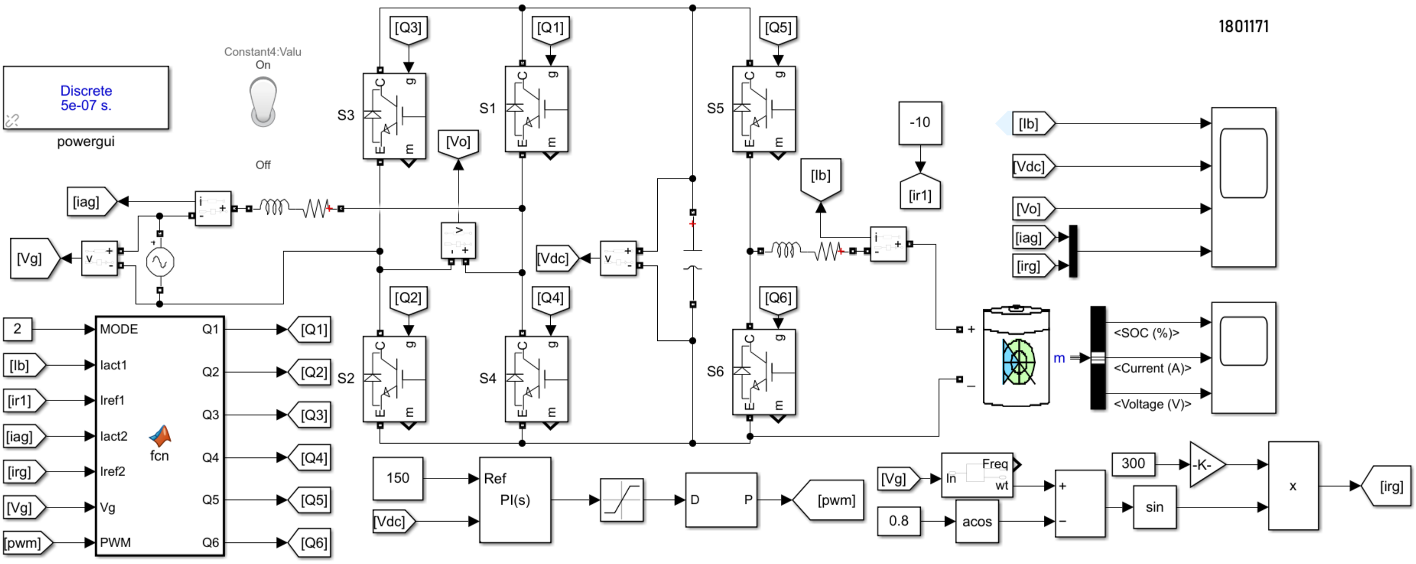


Fig.9.1: Block diagram of bi-directional hysteresis control of grid connected H- bridge and battery with controlled DC linked voltage

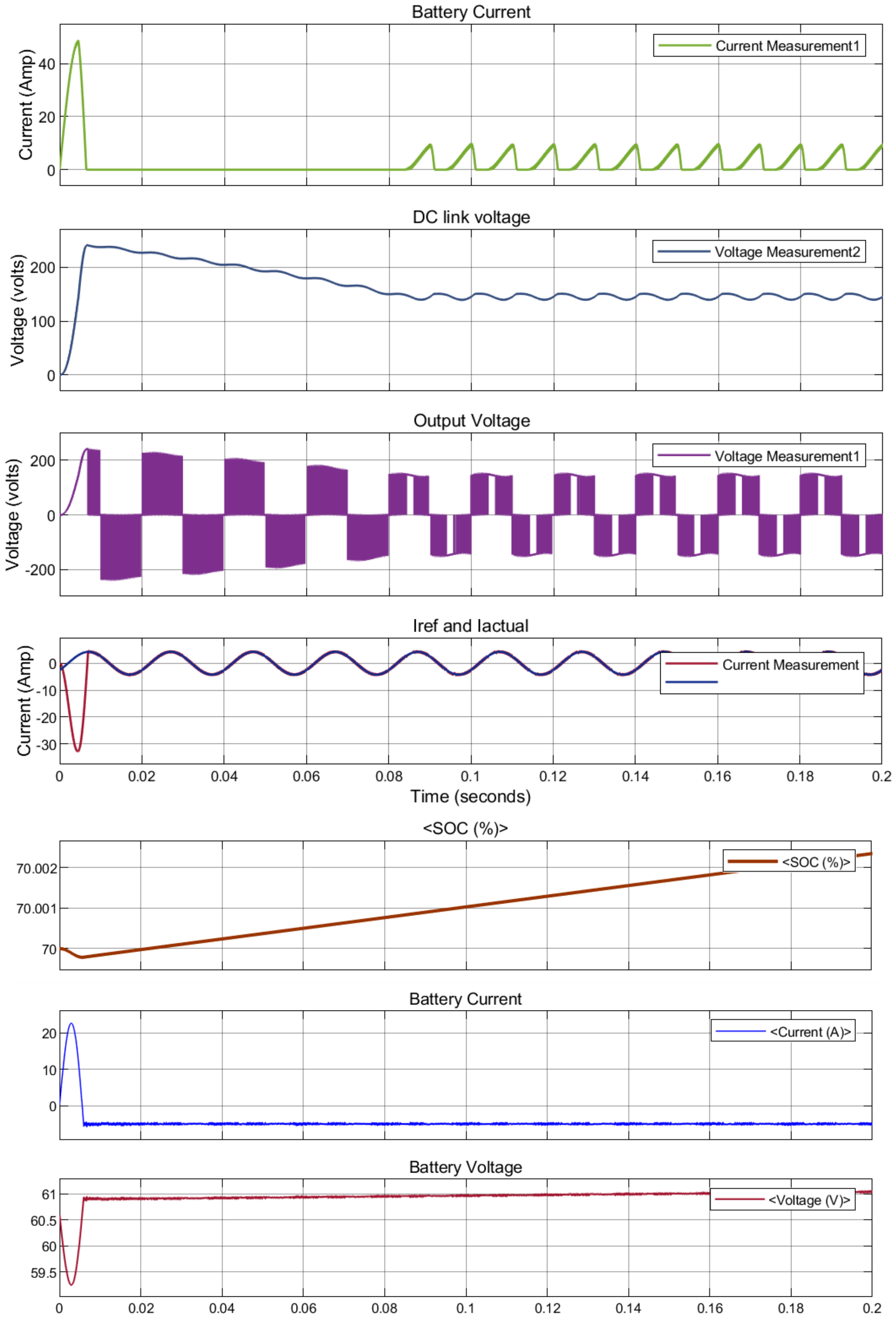


Fig.9.2: Waveform of bi-directional hysteresis control of grid connected H- bridge and battery with controlled DC linked voltage

MATLAB Code

```
function [Q1,Q2,Q3,Q4,Q5,Q6] = fcn(MODE,Iact1,Iref1,Iact2,Iref2,Vg,PWM)
```

```
persistent S1; if isempty(S1);S1=0;end
persistent S2; if isempty(S2);S2=0;end
persistent S3; if isempty(S3);S3=0;end
persistent S4; if isempty(S4);S4=0;end
persistent S5; if isempty(S5);S5=1;end
persistent S6; if isempty(S6);S6=0;end
if MODE==1
    S1=0;S2=0;S3=0;S4=0;S6=0;
    e1=Iact1-Iref1;
    d1=0.5;
    if e1>=d1
        S5=1;
    elseif e1<=-d1
        S5=0;
    else S5=S5;
    end
else MODE==2
    d2=0.5;
    S6=PWM;
    S5=0;
    e2=Iact2-Iref2;
    if Vg>=0
        if e2<=-d2
            S1=1;S2=1;S3=0;S4=0;
        elseif e2>=d2
            S1=1;S2=0;S3=1;S4=0;
        else
            S1 = S1;S2=S2;S3=S3;S4=S4;
        end
    else
        if e2<=-d2
            S1=0;S2=1;S3=0;S4=1;
        elseif e2>=d2
            S1=0;S2=0;S3=1;S4=1;
        else
            S1 = S1;S2=S2;S3=S3;S4=S4;
        end
    end
end
Q1=S1; Q2=S2; Q3=S3; Q4=S4;Q5=S5;Q6=S6;
end
```

9.5 Discussion & Conclusion

This experiment thoroughly investigated hysteresis control of grid connected H bridge system and battery with controlled DC linked voltage. For this system, we utilized connection and value of parameter according to our preference. Similarly, for Hysteresis control inverter, we used necessary tool according to our requirements. Thus, desired output was observed and the simulation was a success.

Experiment No. 10

10.1 Experiment Name

Simulation on Maximum Power Point Tracking (MPPT) algorithm of Photovoltaic system (solar)

10.2 Objectives

- To develop and study a Photovoltaic system (solar) using Simulink
- To get acquainted with Maximum Power Point Tracking (MPPT) algorithm
- To get familiarize with the Simulink platform and Simulink library
- To use the Simulink platform to construct and analyze the I-V and P-V characteristics curve of the system

10.3 Apparatus

- Simulink

10.4 Simulink Block Diagram & Waveform

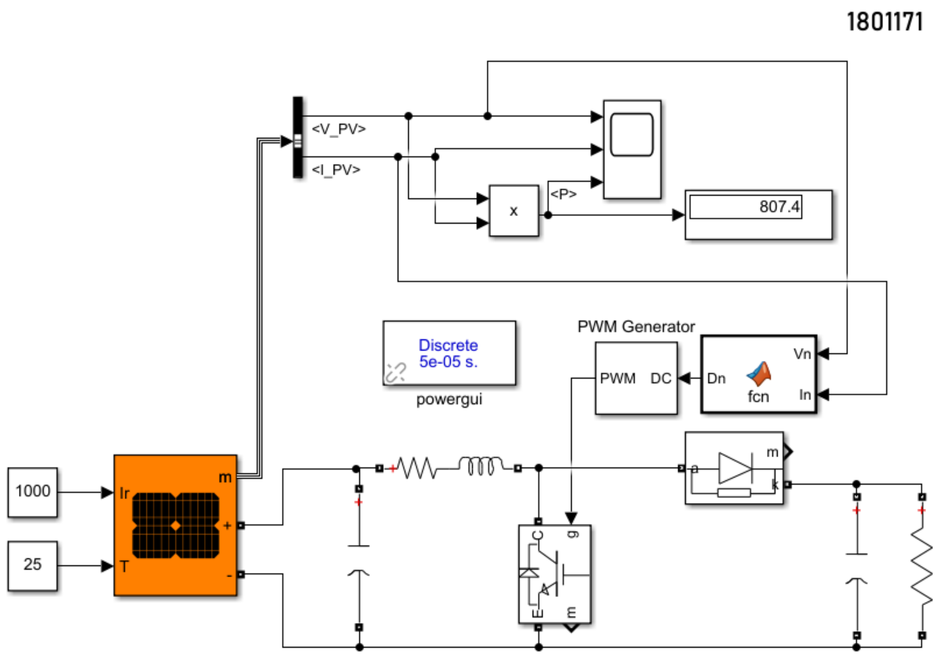


Fig.10.1: Block diagram of a Photovoltaic system (solar) for Maximum Power Point Tracking (MPPT)

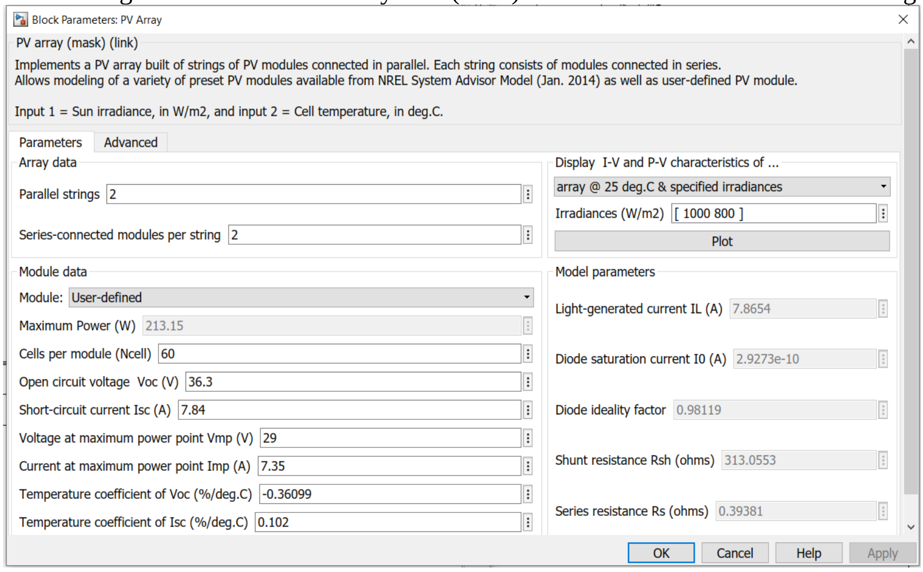


Fig.10.2: Block parameters of the PV array used in Photovoltaic system (solar)

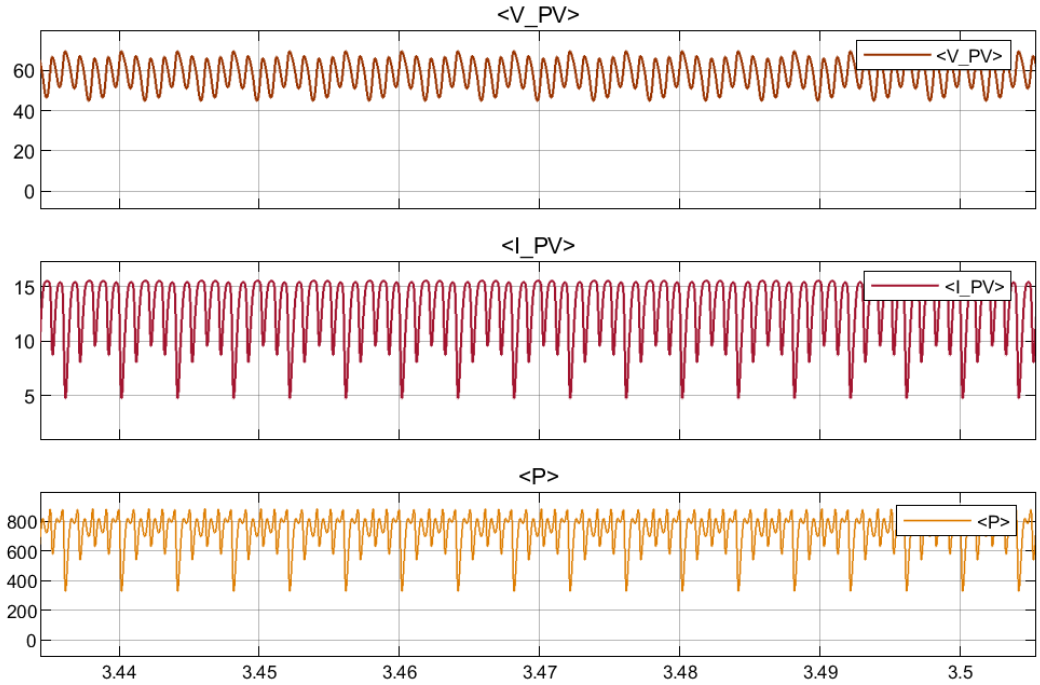


Fig.10.3: Voltage, current, and power waveform of PV array

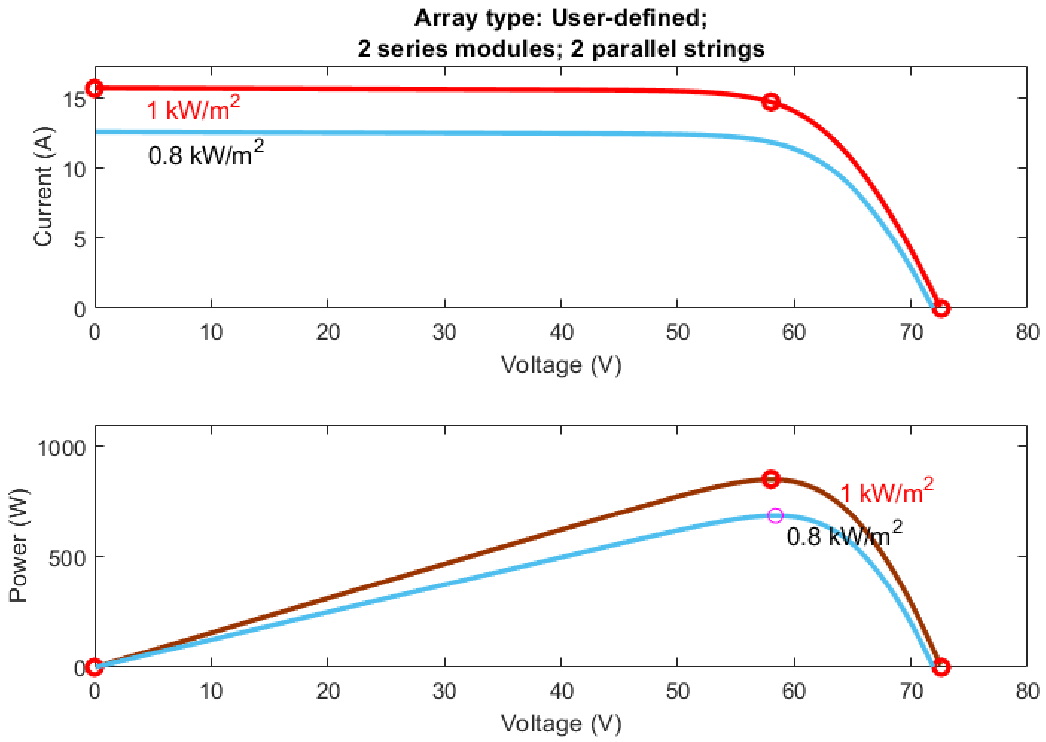


Fig.10.4: I-V and P-V characteristics of defined PV array at 25º Celsius and specified irradiances

MATLAB Code

```
function Dn = fcn(Vn,In)
delta = 0.001;
Dmax = 0.9;
Dmin = 0.05;
persistent Po if isempty(Po); Po = 100; end
persistent Vo if isempty(Vo); Vo = 10; end
persistent Do if isempty(Do); Do = 0.5; end
Pn = Vn*In;
```

```
dp = Pn-Po;
dv = Vn-Vo;
if ((dp/dv)>0)
    Dn = Do-delta;
elseif ((dp/dv)<0)
    Dn = Do+delta;
else
    Dn = Do;
end
Do=Dn;
Po=Pn;
Vo=Vn;
if Dn>Dmax
    Dn = Dmax;
elseif Dn<Dmin
    Dn = Dmin;
else Dn = Dn
end
y = Dn;
end
```

10.5 Discussion & Conclusion

This experiment thoroughly investigated to analyze a Photovoltaic system for Maximum Power Point Tracking (MPPT). Here, we used a solar PV array for analysis. For this system, we utilized connection and value of parameters of the PV array according to our desired preference. This ensured a better characteristics curve for both I-V and P-V. Thus, desired output was observed and the simulation was a success.

<u>Experiment No. 11</u>

11.1 Experiment Name

Simulation on three-phase (abc) to two-phase (αβ0) transformation system

11.2 Objectives

- To develop and perform transformation from three-phase (abc) signal to αβ0 stationary reference frame or the inverse
- To get acquainted with Clark transformation procedure
- To get familiarize with the Simulink platform and Simulink library

11.3 Apparatus

- Simulink

11.4 Simulink Block Diagram & Waveform

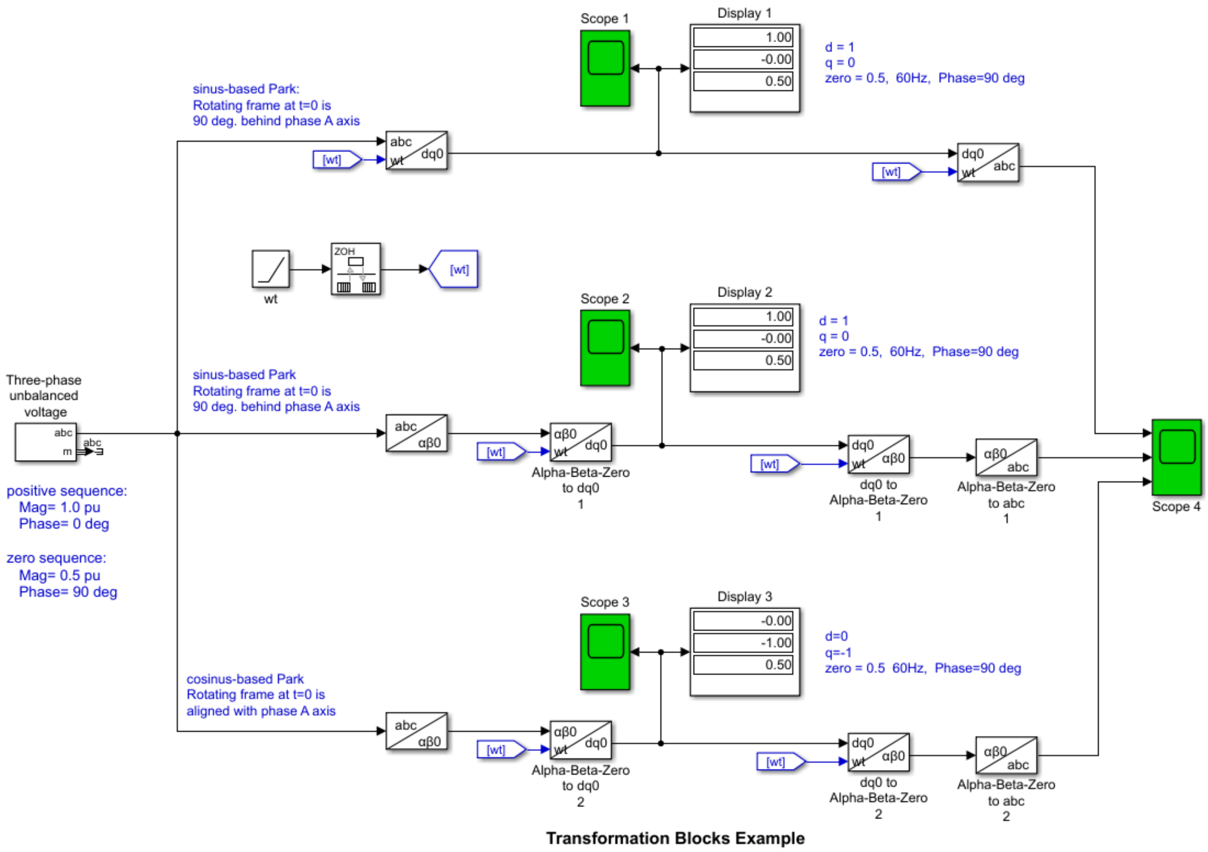
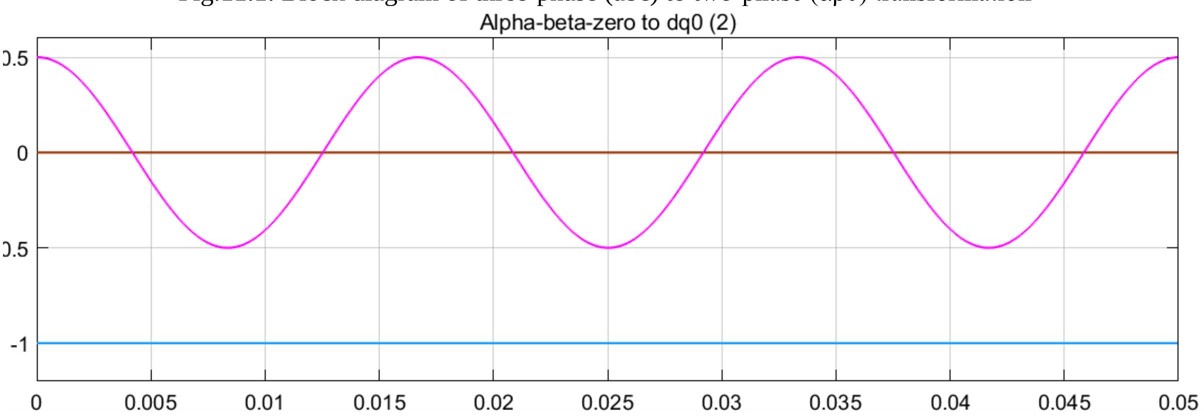


Fig.11.1: Block diagram of three-phase (abc) to two-phase (αβ0) transformation



1801171

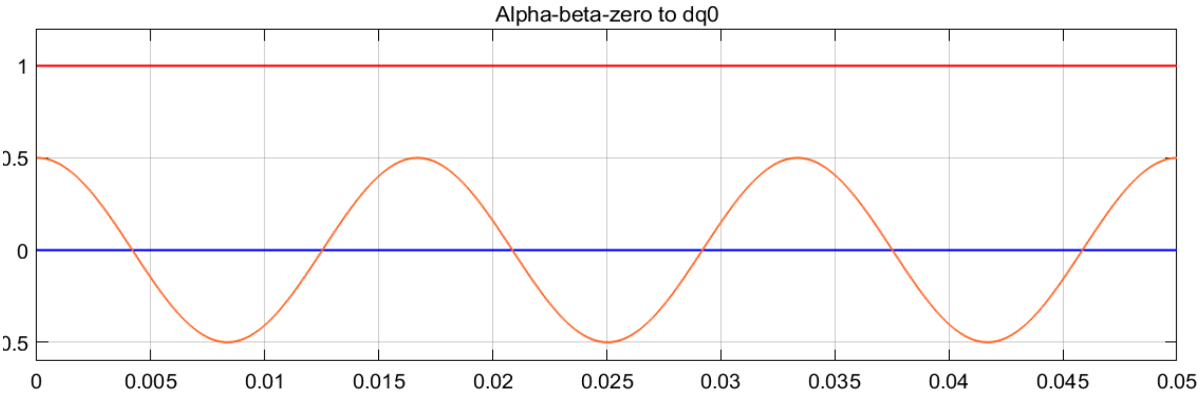


Fig.11.2: Waveform of alpha-beta-zero to dq0

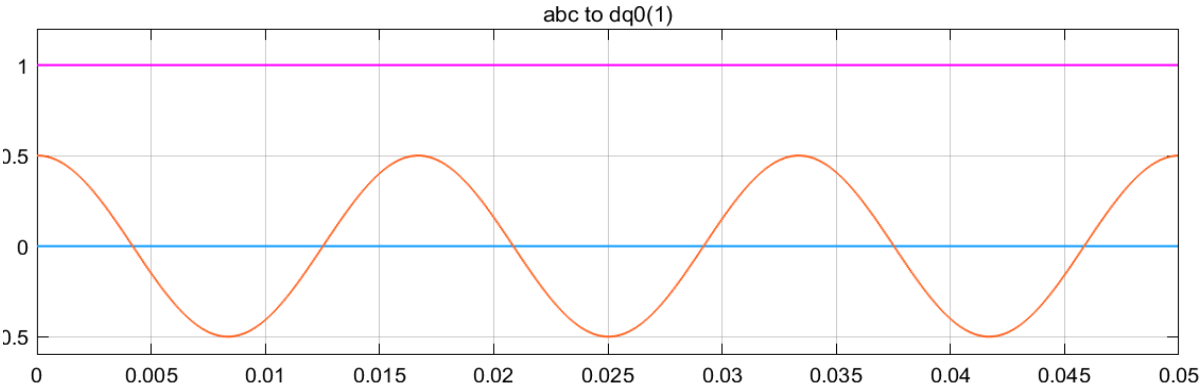


Fig.11.3: Waveform of abc to dq0

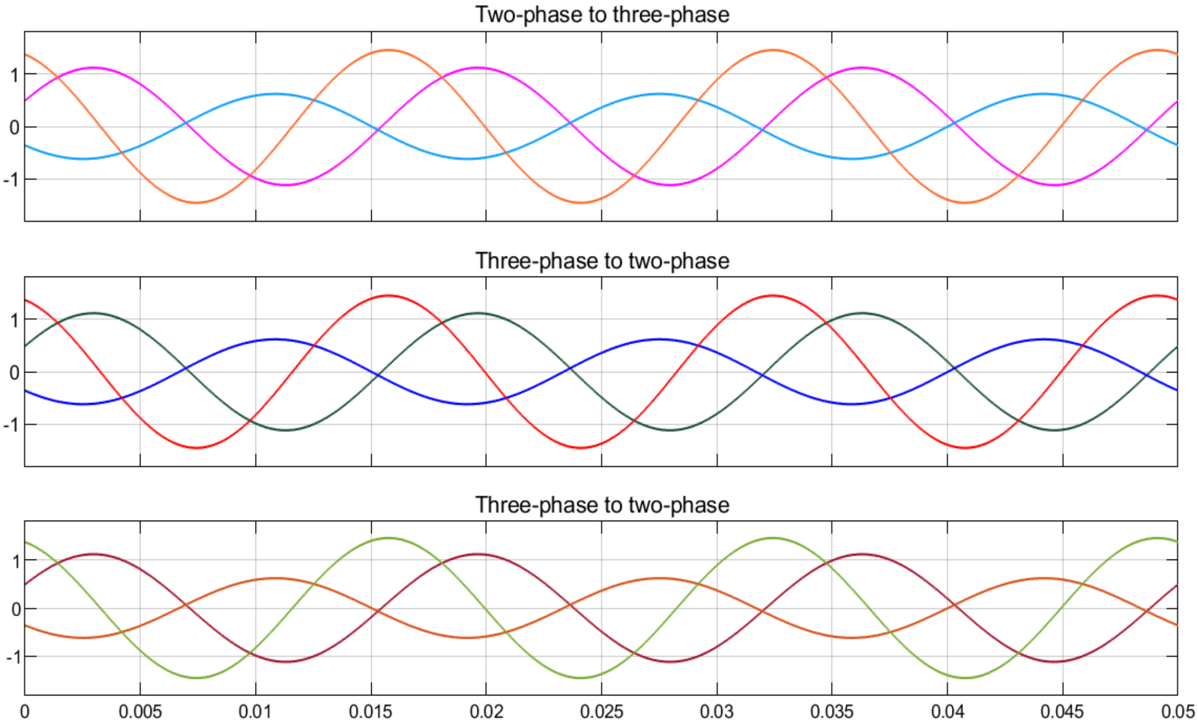


Fig.11.4: Output waveform of three-phase to two-phase transformation and vice versa

11.5 Discussion & Conclusion

This experiment thoroughly investigated to analyze on three-phase (abc) to two-phase (αβ0) transformation system. Here, we used to analyze though Clark transformation procedure to determine theoretical result and compare with the simulated output. This ensured if the experiment carried was accurate or close enough. Thus, desired output was observed and the simulation was a success.