

Single Phase Rectifier with Soft Switching using Pattern Raspberry Pi

Mohamad Afiq Mohd Adha¹, Shamsul Aizam Zulkifli^{1*}

¹Faculty of Electrical and Electronic Engineering,
Universiti Tun Hussein Onn Malaysia, 86400 Parit Raja, Johor, MALAYSIA

*Corresponding Author Designation

DOI: <https://doi.org/10.30880/eeee.2022.03.01.038>

Received 30 January 2022; Accepted 29 Mac 2022; Available online 30 June 2022

Abstract: This project uses the Raspberry Pi model 3 B+ microcontroller to control and build a single-phase rectifier output voltage for soft-switching pattern for Pulse Width Modulation (PWM). The algorithm used in Simulink MATLAB has been applied with the soft-switching outcome takes precedence. The Main feature of this project the Raspberry Pi Microcontroller is to control the whole system that allows to control gate signal of power switches. This project to aim by creating algorithm soft switching in PWM scheme from Simulink MATLAB a MOSFET semiconductor allows for soft- switching pattern in a full wave rectifier. There is one technique used in this project to achieve soft- switching patterns or known is Zero Voltage Switching (ZVS). To make sure to get the desired output voltage with PI controller is used in this system and functionally the output value is equal to the references value and also achieve a minimum error for PWM generation. This is a project to show a generation of soft switching using Rpi microcontroller and designing algorithm of Simulink MATLAB.

Keywords: Single-Phase Rectifier, Soft Switching, PI Controller

1. Introduction

This project has used the soft switching approach to improve conversion efficiency. One approach for permitting a return to a quicker switching frequency. The goal of this project is to create a voltage control strategy using the soft switching approach in order to improve the PWM signal. This technique that involves in this project because it can improve the conversion of efficiency [1], while this technique does not employ any additional resonant parts in the rectifier circuit to shape the switching voltage waveform. Therefore, the Raspberry Pi microcontroller, was constructed using MATLAB Simulink. Then the signal is transmitted to a gate driver circuit to provide a PWM connection to a MOSFET semiconductor bridge rectifier that can control the voltage to eliminate losses.

The hard switching process has the advantage of being relatively quick and easy to implement, but it should be noted that hard switching occurs during the overlap between voltage and current when the MOSFET is switched on and off, which can cause a number of issues such as low switching frequency,

*Corresponding author: aizam@uthm.edu.my

high switching loss, severe turn-on current spikes and turn-off voltage spikes, high electromagnetic interference (EMI), and acoustic noise [2]. Most of the negative impacts of hard switching can be prevented (or at least mitigated) by employing soft switching principles.

In rectifier uses a traditional PWM generation in working principle of PWM generative is to force commutation techniques [3], the authors explained [4] to get DC output voltage result must apply sine wave pulse width modulation (SPWM) that control carrier base modulation technique. In the SPWM technique, the reference current so obtained is phase shifted by an angle (δ). This phase-shifted reference wave is compared with a triangular carrier wave of high frequency to generate the PWM pulses or using hysteresis control. This method is based on the error between the reference output voltage and the measured output voltage. The switching frequency has a direct impact on power dissipation in switching elements such as diodes, transistors, and thyristors, as well as inductive and capacitive parasitic elements and electromagnetic interference (EMI).

2. Methodology

As illustrated in Figure 1, which is a project block diagram, it is an important aspect of the project of development Single-Phase with Rectifier with Soft Switching Pattern using Raspberry Pi. The first stage of this project is MATLAB Simulink software development. The development of PI voltage control and soft switching PWM are also included in this section. The PI controller regulates the voltage sensor's feedback voltage to ensure that the output matches the desired value. The Raspberry Pi as a microcontroller, this board can be used and the data is created in MATLAB Simulink and then been downloaded.

The gate driver block diagram is shown in the third part. This part has one input and one output, with the input connected from the Raspberry Pi and receiving the PWM signal. The output is connected to the rectifier portion, which generates the MOSFET's output voltage, and the other output is connected. A rectifier is a device that converts AC input voltage to DC output voltage. To activate the MOSFET, this circuit received an input signal from a gate driver circuit with a voltage range of 12V.

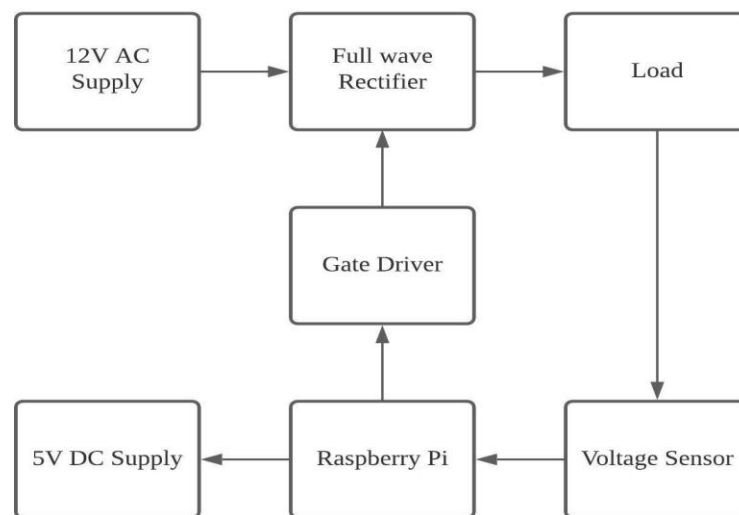


Figure 1: Block Diagram of Single-phase Rectifier with Soft-Switching -(ZVS) pattern PWM using Raspberry Pi

The flow chart for designing soft switching PWM for this project is shown in Figure 2. MATLAB Simulink is used to build the overall circuit of as a simulation system, which includes a single-phase rectifier, filter, load, and PWM signal that is connected to the Soft-Switching signal. At this point, the appropriate filter setting is required to ensure that the output waveform is smooth. The capacitor filter is utilized in this project, and the value of the capacitor is calculated for reducing the voltage ripple.

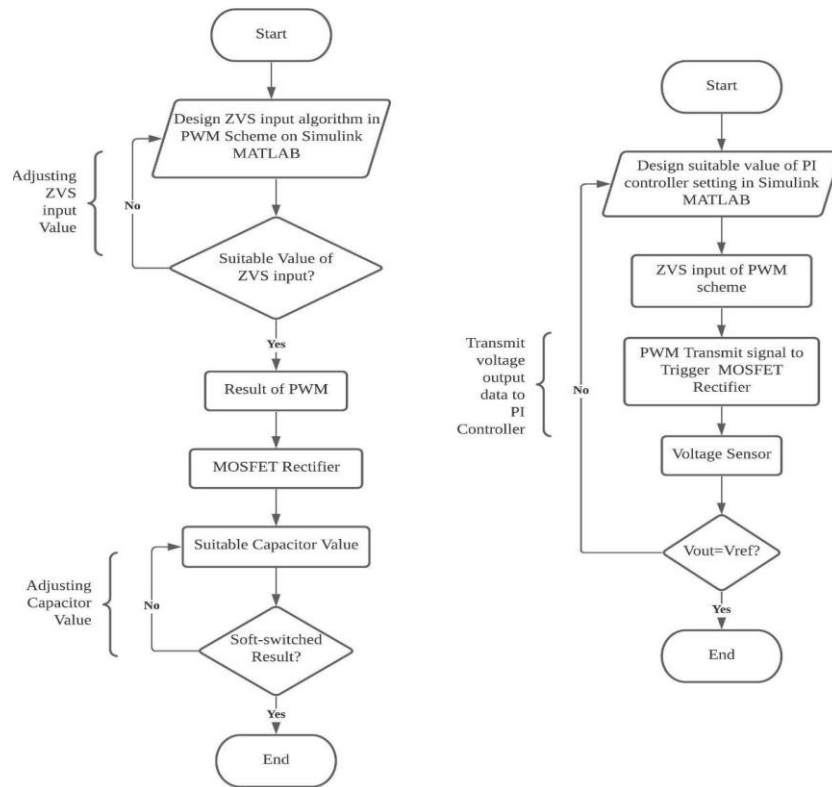


Figure 2: Flowchart of soft switching and PI Controller connected with Single-Phase Rectifier

Figure 3 shows the flow chart for testing Raspberry Pi hardware to communicate with software systems. The algorithm created using MATLAB Simulink is included a PI controller and soft switching algorithm. To connect with the hardware system, MATLAB with the algorithm allows to generate as a signal. The purpose of this software block diagram is to store the signal into the Raspberry Pi board. When the Raspberry Pi hardware receives the signal, it works functionally as a generated PWM soft-switching.

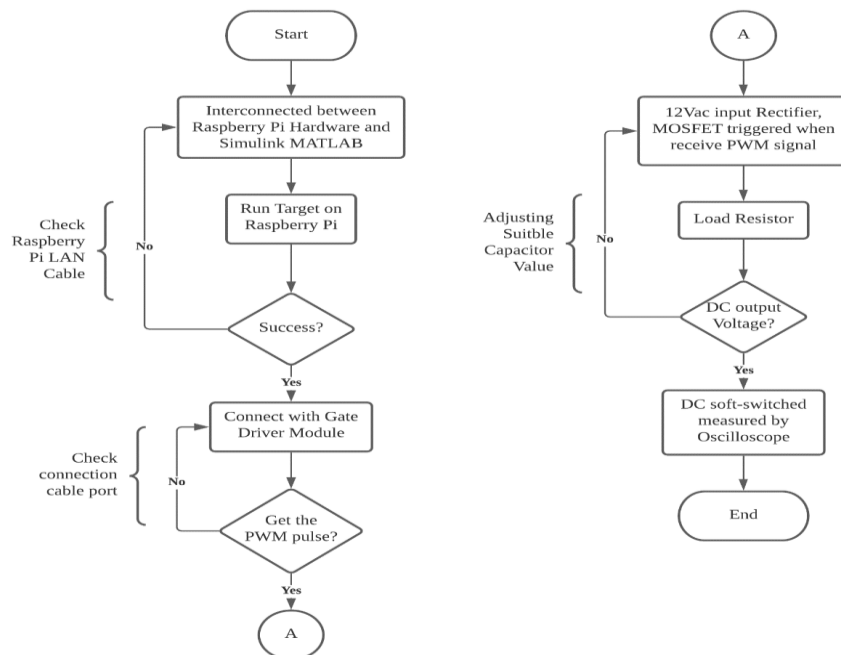


Figure 3: Flowchart of testing Software and Hardware systems communicating

The PI controller block is used as the controller before being uploaded to the Raspberry Pi. The gains for $K_p=100$ and $K_i= 0.1$ as shown in Figure 4. The error, or the difference between measured and reference voltage, with PI control to make the necessary adjustments to reduce the error.

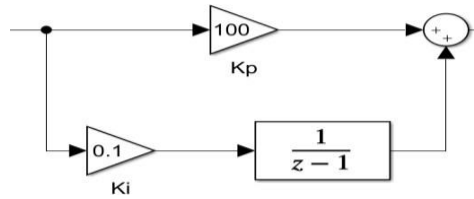


Figure 4: PI Controller

Analog-to-digital (ADC) converter is used in the microcontroller as the input the microcontroller. The main function of this ADC is to convert a signal from analog to digital in discrete form. The analog signal quantities are directly measured, but the digital signal is generally referred to in the form of binary digital [5]. The ADC is created using a block diagram from MATLAB Simulink and downloaded into the Raspberry Pi's microcontroller. The resolution of Raspberry Pi device is 12 bits means can detect the increment voltages. Eq. (1) shows the mathematical of step size to sense voltage. In the meantime, Eq. (2) shows the voltage divider mathematical equation is to allow to step down the voltage.

$$\text{Step Size} = \frac{3.3Vdc}{2^{12}bits} \quad \text{Eq. 1}$$

$$\text{Voltage Divider} = \frac{R_1}{R_1 + R_2} \times Vin \quad \text{Eq. 2}$$

Figure 5 shows single-phase rectifier Simulink MATLAB circuit that has been used to test in open-loop and closed-loop simulations. The structure of soft switching PWM is shown in Figure 6. The OR operator is logically used to combine both the input signal with the switch block, then generates two outputs from PWM to SW₁ and SW₂. It also uses the AND logic operator to combine the input signal with the NOT logic operator to invert the signal from the SW₂. The SW₃ and SW₄ are the same, which links to the delay in obtaining the other negative output condition. The rectifier input source is 12V and is used for converting the signal from the AC signal to the DC signal, enabling the output to be approved as same with theoretical rectifier knowledge.

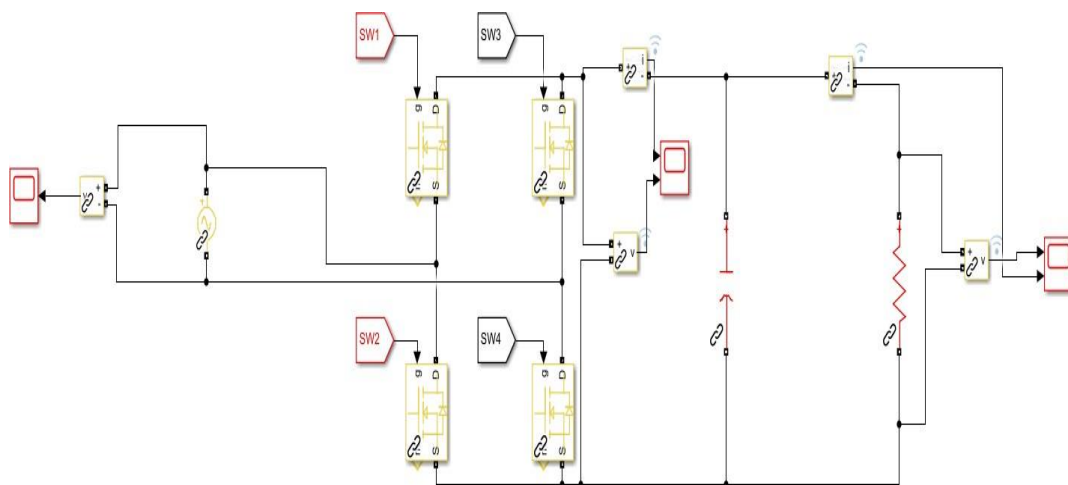


Figure 5: Single-phase Rectifier system Simulink MATLAB

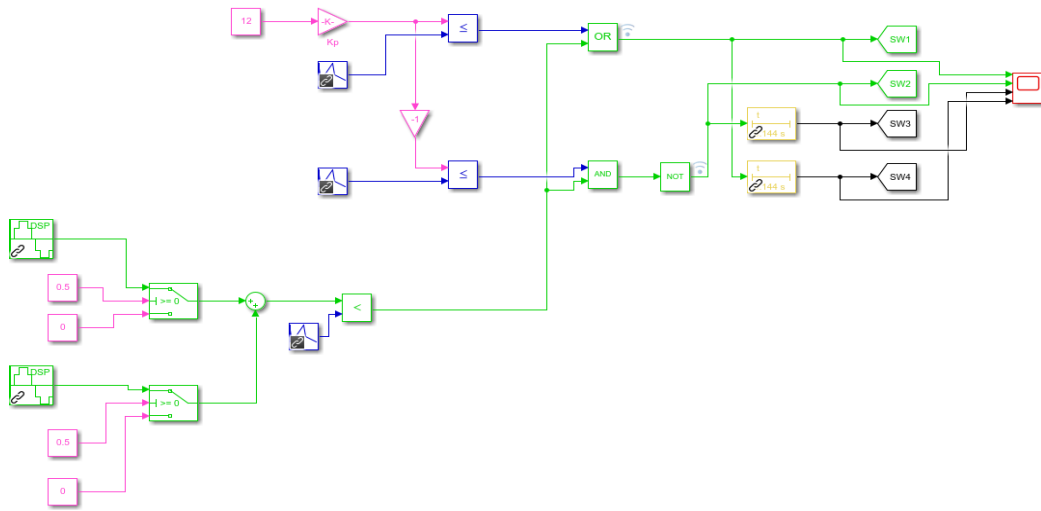


Figure 6: Open Loop system of ZVS PWM Simulink MATLAB

However, in a closed-loop system, as constructed in Simulink MATLAB in Figure 7 with PI controller block which is K_p and K_i to control the output based according to a voltage reference. The load resistor connects with V_{ref} . The PWM is created by injecting the carrier signal into the PI control gain output. The reference voltage value is compared to the voltage feedback output value. The error, or the difference between the reference and measured values, is then error fed into the PI control for be the input to the soft switching PWM operation

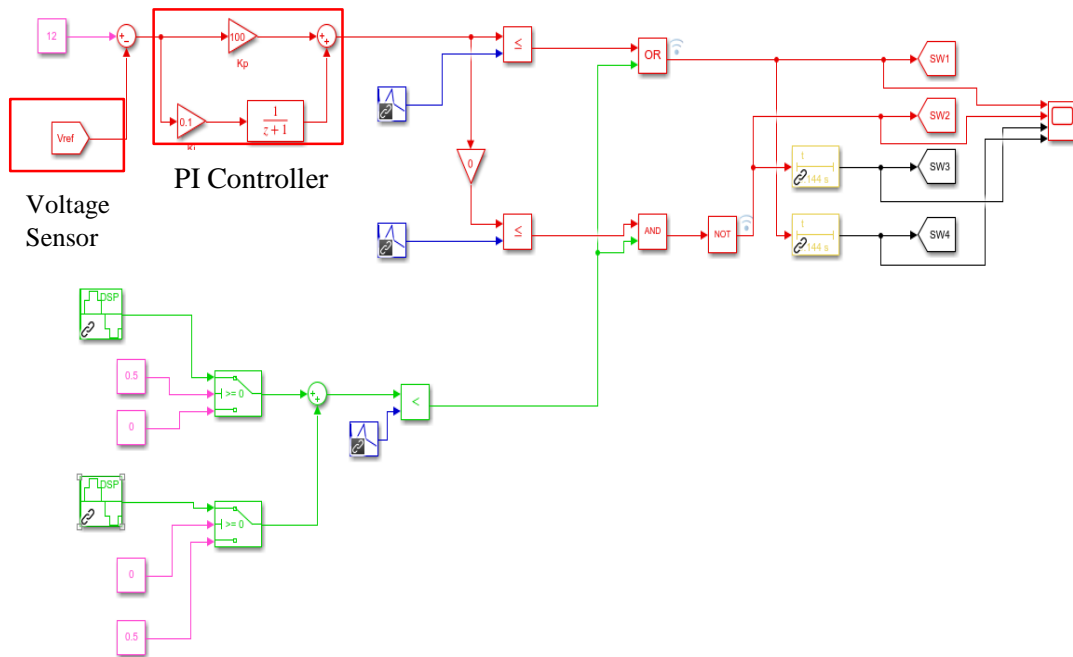


Figure 7: Close Loop system of Soft switching PWM Simulink MATLAB

3. Results and Discussion

In this section, the first test is for the gate driver operation. This microcontroller output only generates 5V, with the gate driver boosting the voltage to 15V for the MOSFET trigger in the rectifier circuit. The output waveform of the gate driver after it boosts the voltage to 14.8Vpk-pk is shown in Figure 8.

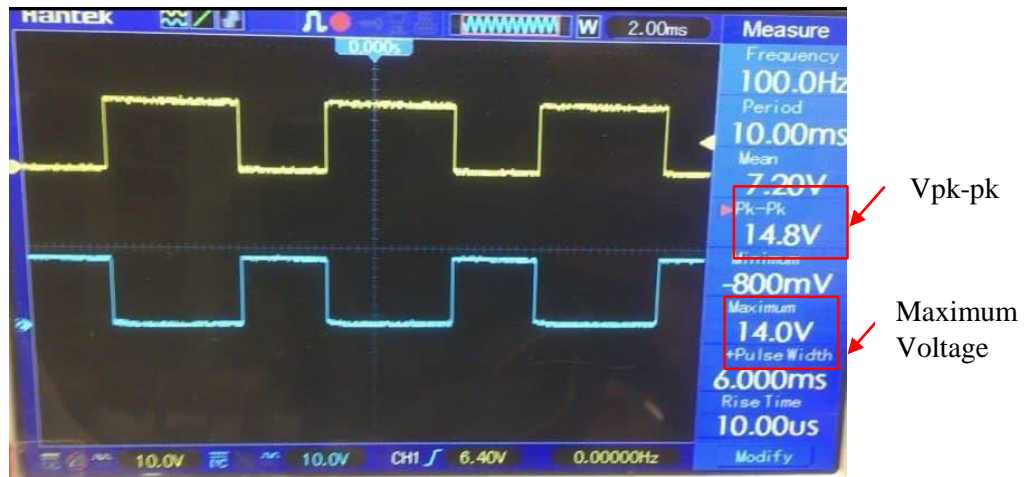


Figure 8: Output from the Gate Driver

Before the hardware is been tested the simulation should be conducted in an open loop system based on Sec.2.4. The result from the simulation the output of the rectifier in shown in Figure 9. The result of soft-switching from the load, the open-loop rectifier output. The value for this simulation on a capacitor which used $4700\mu\text{F}$ and the value of resistor is 10Ω based on Eq. (3) to calculate the V_{ripple} ($V_{\text{pk-pk}}$). This shows a PWM algorithm for a soft-switching signal with Raspberry Pi microcontroller. The voltage maximum reach $12V_{\text{DC}}$.

$$V_r = \frac{1}{(50\text{Hz})(10\Omega)(4700\mu\text{F})} \times 12V_{\text{dc}} = 5.1V_{\text{pk-pk}} \quad (3)$$

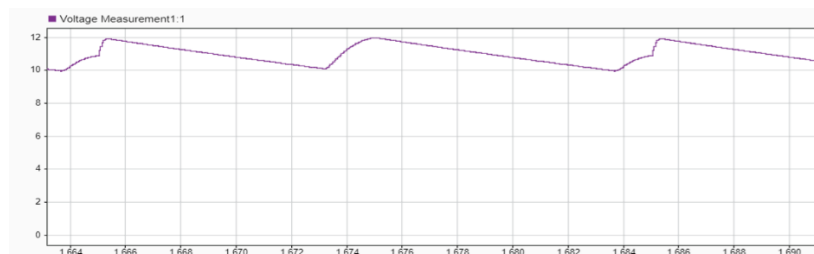


Figure 9: Simulation Result of Output Rectifier Open Loop Systems

The voltage result for open loop rectifier testing after applying the capacitor filter is $3V_{\text{pk-pk}}$ and maximum voltage reach $9.4V_{\text{DC}}$, as shown by the output signal of V_{DC} in Figure 10. The PI controller is work to control the voltage when the V_{ref} is set so that, the output voltage will follow the value of V_{ref} . Figure 11 shows the result simulation when the V_{ref} is set at $10V$ in order to see the PI control is able to track the reference voltage.

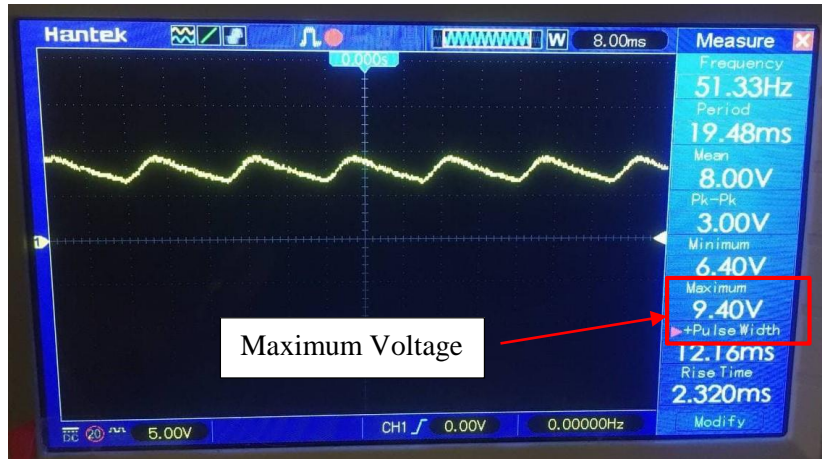


Figure 10: The Output Voltage Waveform of Open Loop System

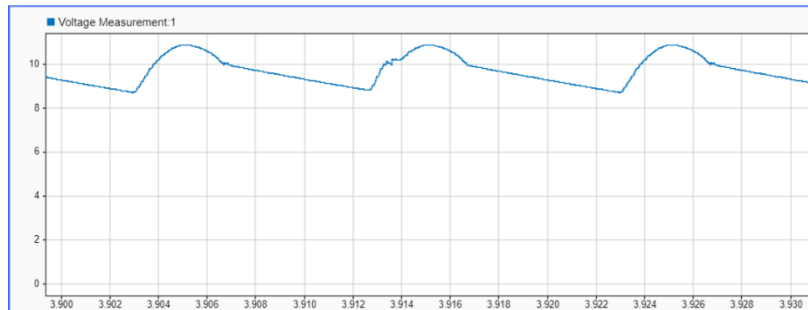


Figure 11: Result of Vref 10V

Figure 12 shows the results on closed loop system with the reference voltage at 10V. The reading of peak to peak is 2.8Vpk-pk and the maximum voltage reach 10V_{DC}. From the observation the output voltage from single phase rectifier is the same with desired output is achieved the target set at the PI simulation before downloaded to the Rpi. As shown in Table 1 the comparison percentage error between actual and simulation result is given where the closed loop results have less error because the PI controller is working.

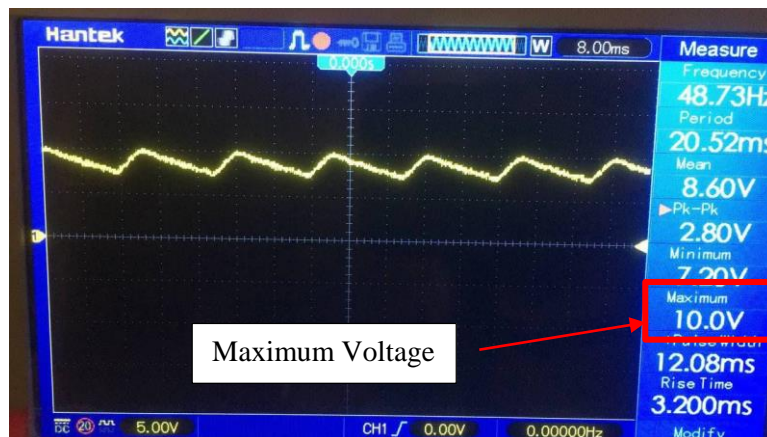


Figure 12: The Output Voltage Waveform of Closed Loop System

Table 1: Comparison between Simulation and Actual in Percentage Error

Type	Simulation Output Voltage Result	Actual Output Voltage Result	Error (%)
Open Loop System	12V	9.4V	21.67%
Closed Loop System	10.8V	10V	7.4%

4. Conclusion

In conclusion, this project used a Raspberry Pi microcontroller to be generated in MATLAB-Simulink. The soft-switching technique is used to reduce overlapping in the switching pattern of PWM and to control voltage that was created in Simulink MATLAB for simulation observation. The PI controller is used to follow the desired output as the result output voltage is controlled for the value maximum voltage. In the end from that, the rectifier output voltage can be controlled by changing based on the value of V_{ref} that is conducted on simulation and hardware setup. However, the closed-loop system uses the available PI controller, ADC block created in Simulink MATLAB and voltage sensor are from voltage divider to allow reduced output voltage to connect with Raspberry Pi. Based on the results and analysis. In the end, the closed-loop hardware has proven that the simulation with the soft switching technique with PI controls is able to respond to the desired target output voltage.

Acknowledgement

The authors would like to thank Advanced Control on Power Converter Team at the Faculty of Electrical and Electronic Engineering, Universiti Tun Hussein Onn Malaysia for its support.

References

- [1] M. K Kazimierzuk, M., & Jozwik, J. (1989). "Class E zero-voltage-switching rectifier with a series capacitor". IEEE Transactions on Circuits and Systems, 36(6), 926-928. doi:10.1109/31.90422.
- [2] Batarseh I., Harb A. (2018) "Soft-Switching dc-dc Converters. In: Power Electronics". Springer, Cham.
- [3] V. Vaideeswaran, N. Sankar, (2018) "Control Technique of Three Phase PWM Rectifiers", International Journal of Engineering and Advanced Technology (IJEAT), Volume 8, ISSN: 2249-8958, December
- [4] Arpit Bohra, Chintan Patel, Divya Sajessh, Micheal Saldana, (2015). "Modulation Technique in Single Phase PWM Rectifier" International Conference on Advances in Science and Technology (ICAST).
- [5] Y. H Dapeng Zhu, Todd Sifleet, T. Nunnally (2007), "Analog to Digital Converters".